

# Domain/Applications Model Layer

## 22. IfcArchitectureDomain

Domain Models, as the name implies, provide a model tailored to the point of view for a particular AEC industry domain or application type. They adapt concepts in the Core and Interoperability layers to this point of view.

The IfcArchitectureDomain schema defines basic object concepts used in Architectural CAD applications that have not been generalized and push lower in the model (e.g. shared with other domains or application types).

### ***Relevant Concepts Modeled Elsewhere in IFC***

#### **1. Concepts modeled in the Core layer schemas**

- Model structuring objects - Project, site, building, building storey, space, building element – all of these fundamental objects, which are shared across all domains in an AEC project are defined in the IFC core model. Please see the containment discussion in the Object Model Guide for discussion of these objects.
- Fundamental properties - Costs, classification, placement, shape, materials and documents – all of these concepts are related to most of the objects used by architects through the object supertypes at in the IFC core model. See notes in the class definitions of this schema for examples where these should be considered.
- Actors – people and/or organizations involved in the project. Actors are important in the IFC model as they concepts such as ownership, responsibility, approval and workflow.
- Assemblies – of elements. This concept is modeled using the objectified relationship IfcRelAssembles. Examples in architecture include stairs, ramps, curtain walls and roofs. The distinction from Nesting is that elements in an assembly can be of various types. Please see the IFC Model Guide for more discussion on this subject.
- Connections – between elements in a project. This concept is modeled in IFC using the objectified relationship IfcRelConnects. Architects are interested many different types of connections because they must design construction details for them. Examples include wall to wall, wall to floor, wall to ceiling and column to beam connections. See IfcRelConnectsElements, its subtypes, IfcConnectionGeometry, its subtypes and the IFC Model Guide for more discussion on this subject.
- Containment – elements that contained in others. This concept is modeled using the objectified relationship IfcRelContains. Examples of interest to architects include the relationships between project, site, building, building storey, spaces and building elements. Please see the IFC Model Guide for more discussion on this subject.
- Controls – conceptual objects that determine or constrain other objects. IFC includes several examples of such controls (constraints, budgets, design program). These are related to the objects which they 'control' through the IfcRelControls objectified relationship. Examples that are of interest to architects include budgets, building code constraints, geometric alignment constraints and space programs (client brief information).
- Groups – of objects, related for some group purpose. Object collections are related to a group object (which defines the purpose) through the IfcRelGroups objectified relationship. Please see the IFC Model Guide for more discussion on this subject.
- Modeling Aids – grids and other concepts which aid in developing a design model. For the architect, the design grid objects are essential. Additionally, reference geometry points, lines and faces are provided as aids in locating design elements. See the Model Guide discussion of the IfcModelingAidExtension schema.
- Nesting – elements that contain other, like elements. This concept is modeled in IFC using the objectified relationship, IfcRelNests. An example of interest to architects is spaces. Architects think of spaces such that they can contain other spaces. That is, they need to be able to 'nest'

spaces inside of other (larger) spaces. Please see the IFC Model Guide for more discussion on this subject.

- Processes and resources – which process products (building elements), resulting in other products (assembly, refined or modified building elements, etc.). The most common examples of are construction processes.
- Proxy objects – surrogates for types of objects that are not yet included in IFC. Proxies are included in IFC in the recognition that IFC will never fully elaborate all of the objects, concepts and processes in the AEC industry. This surrogate allows enables a basic representation of the 'foreign object type' so that the project model is a complete (if not totally accurate) representation. Architects will be most interested in product proxies. Surrogates for real world, physical objects. A shape representation and placement are included for such proxies. This allows architects to understand the shape, location and orientation of the real world object represented by the proxy.
- Runtime assigned properties – properties that are attached to objects depending on a runtime defined "type" or on life cycle stage. For example, some properties only make sense after construction is complete. This concept is handled through use of the objectified relationship `IfcRelAssignsProperties`. Examples of interest to architects include Walls, Doors and Windows – all of which are typed by architects. Generally, this 'typing' is done at some point after conceptual design. That is, the decision about what 'type' of wall, door or window is deferred until the decision is needed. When the 'type' is determined, a number of additional properties can be set. In IFC, these properties are not added to the object until they are needed – by relating one or more 'Property Sets' to the object through the `IfcRelAssignsProperties` relationship object. Please see the IFC Model Guide for more discussion on this subject.
- Space Boundaries – both physical and virtual elements which bound a space. Architects deal extensively with finishes in spaces and often specify such 'interior finishes' for the walls, floor(s) and ceilings that bound a space. IFC includes a special list of relationships from spaces to space boundaries. See `IfcSpace` and `IfcSpaceBoundary` in the `IfcProductExtension` schema.

## 2. Concepts modeled in `IfcSharedBldgElements`

- Walls, doors, windows, columns, beams, floors, roofs – All of these objects types, essential to architectural design are defined in the shared building elements schema because other disciplines also deal with them. Most of these allow specification of 'types' and association of more detailed properties associated with those types.
- Coverings – that cover other building elements. These objects have a special relationship (`IfcRelCoversBldgElements`) to other building elements which they 'cover'. Examples include floor and wall coverings, protective coverings (base molding, chair railing) and ceilings. There is also a special relationship (`IfcRelAttachesToBoundaries`) to space boundaries, which allows specification of finishes by space.
- Curtain walls – assemblies of various elements attached to building structure. Elements of a curtain walls are related to the 'assembly' object through the `IfcRelAssembles` relationship.
- Joints – where two or more building elements come together and 'joined' by more than a connection relationship. Generally, other materials are involved in the construction details for the 'joint.' Examples germane to architects include expansion joints and control joints.
- Screens, louvers, grates and grills – these elements are generalized as permeable opening covers. They can be incorporated into any assembly type (wall, floor, ceiling, curtain wall, etc.) through the `IfcRelAssembles` relationship.

## 3. Concepts modeled in `IfcSharedSpatialElements`

- Fire compartments – this new subtype of `IfcSpace` allows architects to compartmentize buildings in order to meet requirements of fire codes.
- Occupant and Occupancy – architects deal with these concepts in the design and permit phases of projects. Facilities managers use them through the operations phase.
- Space usage profile – this occupancy profile supports design and operation of building systems like HVAC, lighting and shading.

## 4. Concepts modeled in `IfcSharedBldgServiceElements`

- Equipment, electrical appliances – architects deal with many types of equipment. Examples supported in this release of IFC include: elevators, escalators, and window washing equipment.

Architects and interior designers also deal with many types of electrical appliances. Examples supported in this release of IFC include: copiers, phones, facsimiles, computers and printers.

- Fixtures and distribution objects – for ducting, plumbing and electrical systems. Examples of distribution objects supported in this release of IFC (and used by architects) include: ducting, piping, drains, scuppers. Examples of plumbing fixtures used by architects include: faucet, sink, toilet, shower, urinal. Examples of electrical fixtures used by architects include: lights, power outlets and radiant heaters.

#### 5. Concepts modeled in IfcFacilitiesMgmtDomain

- Furniture – both standalone and systems furniture are available in this release of IFC.

## 22.1. Type IfcBuiltInAccessoryTypeEnum

### 22.1.1. Type Semantic Definition

Definition from IFC: Enumeration defining the valid types of Built-In Accessories that can be modeled in this release.

#### History

New Enumeration in IFC Release 2.0

### 22.1.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcBuiltInAccessory

### 22.1.3. Enumeration

DoorOrWindowHardware
PublicRestroom
Unspecified
UserDefined
NotDefined

## 22.2. Type IfcCabinetTypeEnum

### 22.2.1. Type Semantic Definition

Definition from IFC: Enumeration defining the valid types of Built-In cabinets that can be modeled in this release.

#### History

New Enumeration in IFC Release 2.0

### 22.2.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcCabinet

### 22.2.3. Enumeration

Office
--------

Restroom
Storage
Unspecified
UserDefined
NotDefined

## 22.3. Type *IfcCounterOrShelfTypeEnum*

### 22.3.1. Type Semantic Definition

Definition from IFC: Enumeration defining the valid types of Counters/shelves that can be modeled in this release.

#### **History**

New Enumeration in IFC Release 2.0

### 22.3.2. PreDefined Type

This enumeration defines the available PreDefined Types for *IfcCounterOrShelf*

### 22.3.3. Enumeration

CounterTop
Shelf
UserDefined
NotDefined

## 22.4. Type *IfcRailingTypeEnum*

### 22.4.1. Type Semantic Definition

Definition from IFC: Enumeration defining the valid types of Railings that can be modeled in this release.

#### **History**

New Enumeration in IFC Release 2.0

### 22.4.2. PreDefined Type

This enumeration defines the available PreDefined Types for *IfcRailing*

### 22.4.3. Enumeration

Handrail
Guardrail
Balustrade
UserDefined
NotDefined

## 22.5. Type *IfcRampTypeEnum*

### 22.5.1. Type Semantic Definition

Definition from IFC: Enumeration defining the valid types of ramp that can be modeled in this release.

#### History

New Enumeration in IFC Release 2.0

### 22.5.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcRamp

### 22.5.3. Enumeration

Elemented
Layered
Solid
UserDefined
NotDefined

## 22.6. Type *IfcSpaceProgramTypeEnum*

### 22.6.1. Type Semantic Definition

Definition from IFC: This enumeration defines the available Generic Types for IfcSpaceProgram.

#### History

This Enumeration has changed after IFC Release 1.5.1, please see the Migration Guide for details

### 22.6.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcSpaceProgram

### 22.6.3. Enumeration

CirculationSpaceProgram
OccupiedSpaceProgram
OccupiedSpaceProgramStandard
TechnicalSpaceProgram
UserDefined
NotDefined

## 22.7. Type *IfcStairTypeEnum*

### 22.7.1. Type Semantic Definition

Definition from IFC: Enumeration defining the valid types of stair that can be modeled in this release.

### **History**

New Enumeration in IFC Release 2.0

## **22.7.2. PreDefined Type**

This enumeration defines the available PreDefined Types for IfcStair

### **22.7.3. Enumeration**

FireStair
OrnamentalStair
StandardAccessStair
UserDefined
NotDefined

## **22.8. Type IfcVisualScreenTypeEnum**

### **22.8.1. Type Semantic Definition**

Definition from IFC: This enumeration defines the available Generic Types for IfcVisualScreen.

### **History**

New Enumeration in IFC Release 2.0

## **22.8.2. PreDefined Type**

This enumeration defines the available PreDefined Types for IfcVisualScreen

### **22.8.3. Enumeration**

VisualScreenAssembly
VisualScreenDoorOrGate
VisualScreenPost
VisualScreenPanel
VisualScreenRestroomPartition
VisualScreenRestroomPartitionDoor
UserDefined
NotDefined

## **22.9. Class IfcBuiltInAccessory**

### **22.9.1. Class Semantic Definition**

Building hardware or attached occupant accessory - attached to one or more building elements

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- **Document references** – for things like detail drawings, specification sections, cost estimates, etc.  
There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- **Materials** – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set. Three properties in the Pset\_AccessoryCommon allow specification of material, color and finish selections from a manufacturer prescribed list.
- **Assembly** – any other objects considered to be integral to this accessory should be related through the IfcRelAssembles relationship – defining an assembly.

## History

New Entity in IFC Release 2.0

## 22.9.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcBuiltIn
            IfcBuiltInAccessory
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcBuiltInAccessoryTypeEnum	DoorsAndWindowS	CounterOrShelfHW	Bathroom
OPT	calcMountingHeight	height at which the item gets connect to the wall. Value of 0.0 means this property not set.	IfcPositiveLengthMeasure	0	see type	0
OPT	MountingType	Description of the method for mounting	STRING	n/a	n/a	empty string

### Formal Propositions

WR61	
------	--

## 22.9.3. Interface Definitions

I\_BuiltInAccessory

## 22.9.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
DoorOrWindowHardware	Pset_AccessoryDoorOrWindowHardware
PublicRestroom	Pset_AccessoryPublicRestroom
Unspecified	Pset_AccessoryUnspecified
UserDefined	
NotDefined	

## 22.9.5. Geometry Use Definitions

### **Context for Geometric Representations**

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### **Reference Geometry**

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### **Geometry Representations:**

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## 22.10. Class IfcCabinet

### 22.10.1. Class Semantic Definition

Storage enclosure, normally attached to a wall and/or floor. Typically includes doors and internal shelves.

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- Materials – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set.
- Assembly – any other objects like rails, brackets, enclosed power boxes, lights, etc., considered to be integral to this cabinet should be related through the IfcRelAssembles relationship – defining a cabinet assembly.

### **History**

New Entity in IFC Release 2.0



## 22.10.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcBuiltIn
            IfcCabinet
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcCabinetTypeEnum	Bathroom	Office	Bathroom
	CabinetHardware	List of references to accessory hardware for this cabinet.	LIST [0:?] OF IfcBuiltInAccessory	n/a	n/a	empty list

### Formal Propositions

WR71	
------	--

## 22.10.3. Interface Definitions

I\_Cabinet

## 22.10.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
Office	Pset_CabinetOffice
Restroom	Pset_CabinetRestroom
Storage	Pset_CabinetStorage
Unspecified	Pset_CabinetUnspecified
UserDefined	
NotDefined	

## 22.10.5. Geometry Use Definitions

### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### Reference Geometry

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### **Geometry Representations:**

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## **22.11. Class IfcCounterOrShelf**

### **22.11.1. Class Semantic Definition**

Horizontal work or storage surface attached to a wall or covering the top of a cabinet.

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- **Document references** – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- **Materials** – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set.
- **Assembly** – any other objects like backslashes, beams, support brackets and rails, built-in power boxes, built-in appliances, etc., considered to be integral to this counter or shelf should be related through the IfcRelAssembles relationship – defining a counter or shelf assembly.

### **History**

New Entity in IFC Release 2.0

### **22.11.2. Attribute and Relationship Definitions**

#### **Superclasses and Subclasses**

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcBuiltIn
            IfcCounterOrShelf
  
```

#### **Attributes and Relationships**

Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
----------------------	------------	-------------------	------	------	---------

	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcCounterOrShelfTypeEnum	BathroomCounter	Shelf	Shelf
	CounterOrShelfHardware	List of references to accessory hardware for this counter or shelf.	LIST [0:?] OF IfcBuiltInAccessory	n/a	n/a	empty list

### Formal Propositions

WR71	
------	--

## 22.11.3. Interface Definitions

I\_CounterOrShelf

## 22.11.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
CounterTop	Pset_Counter
Shelf	Pset_Shelf
UserDefined	
NotDefined	

## 22.11.5. Geometry Use Definitions

### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### Reference Geometry

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### Geometry Representations:

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### Standard 3D Geometric Representation

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### Advanced 3D Geometric Representation

There is no advanced geometry representation defined for this object type in this release.

### Arbitrary 3D Geometric Representation

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## 22.12. Class IfcLanding

### 22.12.1. Class Semantic Definition

Floor section to which one or more stair flights or ramp flights connects. May or may not be adjacent to a building storey floor.

#### Relevant Concepts Modeled Elsewhere in IFC

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- **Document references** – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- **Materials** – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set.
- **Assembly** – any supporting structural elements (beams, joists, etc.) considered to be integral to this landing should be related through the IfcRelAssembles relationship – defining a landing assembly. Railings and connected stair or ramp flights will be related through an overall stair or ramp assembly (see those classes for more discussion).

#### History

New Entity in IFC Release 2.0

### 22.12.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcSlab
            IfcLanding
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	calcHeadRoom	Headroom clearance	IfcPositiveLengthMeasure	0	see type	0
OPT	calcWidth	Width of this landing	IfcPositiveLengthMeasure	0	see type	0
OPT	calcLength	Length of this landing (direction of travel)	IfcPositiveLengthMeasure	0	see type	0

### 22.12.3. Interface Definitions

I\_StairOrRampLanding

### 22.12.4. Geometry Use Definitions

#### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric

representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### **Reference Geometry**

The definition of the object coordinate system for this object type is defined in its supertype `IfcProduct`. It is defined by the following:

- `IfcLocalPlacement` -- which defines the local coordinate system that is referenced by all geometric representations.

### **Geometry Representations:**

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (`IfcFacetedBrepWithVoids`) or exclude them (`IfcFacetedBrep`).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## **22.13. Class *IfcRailing***

### **22.13.1. Class Semantic Definition**

Frame assembly adjacent to human circulation spaces and at some space boundaries where in lieu of walls or to compliment walls. Designed to aid humans, either as an optional physical support, or to prevent injury by falling.

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at `IfcObject.DocumentReferences` (a supertype for this object type) Set 'DocumentPurpose' on the `IfcDocumentReference` object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- Materials – see `IfcBuildingElement.HasMaterial`. This is a material select, which supports an individual material, a list of materials or a material layer set. The `MaterialList` should be used for this object type. The property 'RailingMaterial' in the `Pset_RailingCommon` provides an integer index into this material list to indicate the material for the railing stiles. The property 'HandrailMaterial' in `Pset_RailingHandrail` provides an integer index to indicate the handrail material.
- Assembly – Associated brackets, anchors, posts, beams, ornamental attachments or other objects considered to be integral to this railing should be related through the `IfcRelAssembles` relationship – defining a railing assembly.

### **History**

New Entity in IFC Release 2.0

## 22.13.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcRailing

```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcRailingTypeEnum	Handrail	Balustrade	Handrail
	RailingHardware	List of references to accessory/mounting hardware for this railing.	LIST [0:?] OF IfcBuiltInAccessory	n/a	n/a	empty list

### Formal Propositions

WR61	
------	--

## 22.13.3. Interface Definitions

I\_Railing

## 22.13.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
Handrail	Pset_RailingHandrail
Guardrail	Pset_RailingGuardrail
Balustrade	Pset_RailingBalustrade
UserDefined	
NotDefined	

## 22.13.5. Geometry Use Definitions

### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### Reference Geometry

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### ***Geometry Representations:***

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### ***Standard 3D Geometric Representation***

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### ***Advanced 3D Geometric Representation***

There is no advanced geometry representation defined for this object type in this release.

### ***Arbitrary 3D Geometric Representation***

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## ***22.14. Class IfcRamp***

### **22.14.1. Class Semantic Definition**

An assembly of IfcRampFlight, IfcLanding, IfcRailing and other objects which provide a human circulation link between different slabs (floors, landings, walkways, etc.) in a project.

### ***Relevant Concepts Modeled Elsewhere in IFC***

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like detail drawings, specification sections, cost estimates, etc.  
There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- Materials – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set. The MaterialList should be used for this object type as is will almost always involve multiple materials.
- Assembly – Ramp flights, landings, railings, supporting structural elements (beams, joists, etc.), or other objects considered to be part of this ramp should be related through the IfcRelAssembles relationship – defining the ramp assembly.

### ***History***

New Entity in IFC Release 2.0

### **22.14.2. Attribute and Relationship Definitions**

#### ***Superclasses and Subclasses***

```
graph TD
    IfcRoot --> IfcObject
    IfcObject --> IfcProduct
    IfcProduct --> IfcElement
    IfcElement --> IfcBuildingElement
    IfcBuildingElement --> IfcRamp
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcRampTypeEnum	Elemented	Solid	Layered
	VerticallyConnects		LIST [0:?] OF IfcSlab			

### Formal Propositions

WR61	
------	--

## 22.14.3. Interface Definitions

I\_Ramp

## 22.14.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
Elemented	Pset_RampElemented
Layered	Pset_RampLayered
Solid	Pset_RampSolid
UserDefined	
NotDefined	

## 22.14.5. Geometry Use Definitions

### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### Reference Geometry

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### Geometry Representations:

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### Standard 3D Geometric Representation

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### Advanced 3D Geometric Representation

There is no advanced geometry representation defined for this object type in this release.



## Arbitrary 3D Geometric Representation

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## 22.15. Class IfcRampFlight

### 22.15.1. Class Semantic Definition

Inclined slab segment, normally providing a human circulation link between two landings, floors or slabs at different elevations.

#### Relevant Concepts Modeled Elsewhere in IFC

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- Materials – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set. The MaterialList should be used for this object type as is will almost always involve multiple materials.
- Assembly – any supporting structural elements (beams, joists, etc.) considered to be integral should be related through the IfcRelAssembles relationship – defining an assembly.

#### History

New Entity in IFC Release 2.0

### 22.15.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcRampFlight
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	VerticallyConnects		LIST [0:2] OF IfcSlab			
OPT	calcLength	length of ramp	IfcPositiveLengthMeasure	0	see type	0
OPT	calcWidth	width of ramp	IfcPositiveLengthMeasure	0	see type	0
OPT	calcRise	rise of ramp	IfcPositiveLengthMeasure	0	see type	0
OPT	calcSlope	slope of ramp - relative to horizontal (non-sloping) floor	IfcPlaneAngleMeasure	0	see type	0

### 22.15.3. Interface Definitions

I\_RampFlight

## 22.15.4. Geometry Use Definitions

### **Context for Geometric Representations**

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### **Reference Geometry**

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### **Geometry Representations:**

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## 22.16. Class IfcRelAdjacencyReq

### 22.16.1. Class Semantic Definition

Objectified Relationship defines requirements for the adjacency of two spaces in the architectural program. The Adjacency required is encoded as an integer value between 0 and 256, where 0 means the spaces must be immediately adjacent and 256 means that they should be as far apart as possible.

ISSUES: No issues to date.

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- None specified at this time.

### **History**

This Entity has changed after IFC Release 1.5.1, please see the Migration Guide for details

## 22.16.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

IfcRoot  
IfcRelationship  
**IfcRelAdjacencyReq**

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	RelatingSpaceProgram	Architectural program for the first Space.	IfcSpaceProgram	n/a	n/a	NIL
	RelatedSpaceProgram	Architectural program for the second Space.	IfcSpaceProgram	n/a	n/a	NIL
	AdjacencyImportanceRating	Integer value (between 0 and 256) for the required adjacency between these two spaces. 0=immediate adjacency required, 256=spaces should be as far apart as possible.	INTEGER	0	256	0

## 22.16.3. Interface Definitions

I\_RelAdjacencyReq

## 22.16.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 22.17. Class IfcSpaceProgram

### 22.17.1. Class Semantic Definition

*Definition from IFC:* Architectural program for a space in the building or facility being designed; essentially the requirements definition for such a building space. Note that this 'program' defined the client requirements for the space before the building is designed. Space programs can change over the life cycle of a building, after the building is occupied. Changes to space programs take place in the facilities management/operations phase of the building life cycle.

#### Relevant Concepts Modeled Elsewhere in IFC

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like client briefing documents, conceptual space drawings, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Client brief", "Conceptual space drawing", etc.).

#### History

This Entity has changed after IFC Release 1.5.1, please see the Migration Guide for details

## 22.17.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcControl
      IfcSpaceProgram
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	SpaceProgramName		STRING			
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcSpaceProgramTypeEnum	CirculationSpaceProgram	TechnicalSpaceProgram	OccupiedSpaceProgram
INV	HasAdjacencyReqsTo	Set of inverse relationships to Space adjacency objects (FOR RelatingObject).	SET [0:?] OF IfcRelAdjacencyReq	n/a	n/a	NIL
INV	HasAdjacencyReqFrom	Set of inverse relationships to Space adjacency objects (FOR RelatedObject).	SET [0:?] OF IfcRelAdjacencyReq	n/a	n/a	NIL

### Formal Propositions

WR41	
------	--

## 22.17.3. Interface Definitions

I\_SpaceProgram

## 22.17.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
CirculationSpaceProgram	Pset_SpaceProgramCirculation
OccupiedSpaceProgram	Pset_SpaceProgramOccupied
OccupiedSpaceProgramStandard	Pset_SpaceProgramOccupiedStandard
TechnicalSpaceProgram	Pset_SpaceProgramTechnical
UserDefined	
NotDefined	

## 22.17.5. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 22.18. Class IfcSpaceProgramGroup

### 22.18.1. Class Semantic Definition

*Definition from IFC:* A collection of building spaces that will be used by a single functional group within the occupying organization.

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like client briefing documents, conceptual space drawings, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Client brief", "Conceptual space drawing", etc.).

### **History**

This Entity has changed after IFC Release 1.5.1, please see the Migration Guide for details

## **22.18.2. Attribute and Relationship Definitions**

### **Superclasses and Subclasses**

```

IfcRoot
  IfcObject
    IfcGroup
      IfcSpaceProgramGroup
  
```

### **Attributes and Relationships**

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	RequiredGroupArea	Total area required by this group. NOTE: this should be satisfied by the list of associated spaces, but may not be.	IfcAreaMeasure	0	see type	0
OPT	GroupRole	Role of this group in the Program	STRING	n/a	n/a	empty string
OPT	GroupAssignment	Definition of an individual or organization in the Architectural Program	IfcActorSelect	n/a	n/a	NIL

### **Formal Propositions**

WR41	Ensure that only space programs (IfcSpaceProgram) get grouped by virtue of the general grouping mechanism (IfcRelGroups).
------	---

## **22.18.3. Interface Definitions**

I\_SpaceProgramGroup

## **22.18.4. Geometry Use Definitions**

Instances of this class have no physical presence and therefore no geometric representation.

## **22.19. Class IfcStair**

### **22.19.1. Class Semantic Definition**

Assembly of building components allowing occupants to walk (step) from Floor (or Landing) to another at a different elevation.

## Relevant Concepts Modeled Elsewhere in IFC

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- **Document references** – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- **Materials** – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set. The MaterialList should be used for this object type as is will almost always involve multiple materials.
- **Assembly** – Stair flights, landings, railings, supporting structural elements (beams, joists, etc.), or other objects considered to be part of this stair should be related through the IfcRelAssembles relationship – defining the stair assembly.

## History

New Entity in IFC Release 2.0

## 22.19.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcStair
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcStairTypeEnum	FireStair	Standard AccessStair	Standard AccessStair
	VerticallyConnects	List of Floors to which this stair assembly connects. Through these relationships, one can determine which building storey's are served by this stair	LIST [0:?] OF IfcSlab	n/a	n/a	empty list

### Formal Propositions

WR61	
------	--

## 22.19.3. Interface Definitions

I\_Stair

## 22.19.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
FireStair	Pset_StairFire
OrnamentalStair	Pset_StairOrnamental
StandardAccessStair	Pset_StairAccess

UserDefined	
NotDefined	

## 22.19.5. Geometry Use Definitions

### **Context for Geometric Representations**

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### **Reference Geometry**

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### **Geometry Representations:**

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## 22.20. Class IfcStairFlight

### 22.20.1. Class Semantic Definition

Assembly of building components in a single "run" of stair steps (not interrupted by a landing). The Stair steps and any stringers are included in this object.

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like detail drawings, specification sections, cost estimates, etc. There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- Materials – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set. The MaterialList should be used for this object type as is will almost always involve multiple materials.

- Assembly – any supporting structural elements (beams, joists, etc.) considered to be integral should be related through the IfcRelAssembles relationship – defining an assembly.

### History

New Entity in IFC Release 2.0

## 22.20.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcStairFlight

```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	VerticallyConnects	List of relationships - Floors or Stair Landings (subtype of floor) to which this flight connects.	LIST [0:2] OF IfcSlab	n/a	n/a	empty list
OPT	StepTreadMaterial	Building material used in the step treads	IfcMaterial	n/a	n/a	NIL
OPT	StepNosingMaterial	Building material used in the step nosings	IfcMaterial	n/a	n/a	NIL
OPT	calcStepRise	Vertical distance from tread to tread	IfcPositiveLengthMeasure	0.0	300 mm	200 mm
OPT	calcStepTread	Horizontal distance from the front to the back of the tread	IfcPositiveLengthMeasure	0.0	see type	300 mm
OPT	calcFlightHeadRoom	Headroom clearance for this flight	IfcPositiveLengthMeasure	0.0	see type	1 cm
OPT	calcTotalFlightRise	Total "rise" in this stair flight assembly	IfcPositiveLengthMeasure	0.0	see type	1 cm
OPT	calcTotalFlightRun	Total "run" in this stair flight assembly	IfcPositiveLengthMeasure	0.0	see type	1 cm

## 22.20.3. Interface Definitions

I\_StairFlight

## 22.20.4. Geometry Use Definitions

### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### Reference Geometry

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.



### **Geometry Representations:**

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## **22.21. Class IfcVisualScreen**

### **22.21.1. Class Semantic Definition**

Physical barrier to block visual connection. An element or assembly whose purpose is to "screen" an area from human view

### **Relevant Concepts Modeled Elsewhere in IFC**

For concepts relevant to architecture, but not specific to this class, see the 'Relevant Concepts' section at the beginning of this schema.

- Document references – for things like detail drawings, specification sections, cost estimates, etc.  
There is a list of such references at IfcObject. DocumentReferences (a supertype for this object type) Set 'DocumentPurpose' on the IfcDocumentReference object to declare the purpose of the referenced document (e.g. "Construction detail", "Construction specification", etc.).
- Materials – see IfcBuildingElement.HasMaterial. This is a material select, which supports an individual material, a list of materials or a material layer set. Three properties in the Pset\_VisualScreenCommon allow specification of material, color and finish selections from a manufacturer prescribed list.
- Assembly – Visual screen posts, panels, doors, gates and other objects considered to be part of a Visual Screen object typed as "VisualScreenAssembly" should be related through the IfcRelAssembles relationship – defining the visual screen assembly.

### **History**

New Entity in IFC Release 2.0

### **22.21.2. Attribute and Relationship Definitions**

#### **Superclasses and Subclasses**

```
IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcVisualScreen
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Determines which type defining PropertySet will be attached to this object	IfcVisualScreenTypeEnum	ScreenAssembly	ScreenDoorOrGate	ScreenPanel

### Formal Propositions

WR61	
------	--

## 22.21.3. Interface Definitions

I\_VisualScreen

## 22.21.4. Type Definitions

### Common PropertySet

Pset\_VisualScreenCommon

### Type driven PropertySets

PreDefined Type	Associated PropertySet
VisualScreenAssembly	Pset_VisualScreenAssembly
VisualScreenDoorOrGate	Pset_VisualScreenDoorOrGate
VisualScreenPost	Pset_VisualScreenPost
VisualScreenPanel	Pset_VisualScreenPanel
VisualScreenRestroomPartition	Pset_VisualScreenRestroomPartition
VisualScreenRestroomPartitionDoor	Pset_VisualScreenRestroomPartitionDoor
UserDefined	
NotDefined	

## 22.21.5. Geometry Use Definitions

### Context for Geometric Representations

The geometric representation of this object type is given by one or more of the 'Representations' on the IfcProduct supertype. Specifically, those of type IfcProductDefinitionShape. Multiple geometric representations are supported. These representations are coordinated by a shared reference geometry. For this object type the referenced geometry consists of a local placement only.

### Reference Geometry

The definition of the object coordinate system for this object type is defined in it's supertype IfcProduct. It is defined by the following:

- IfcLocalPlacement -- which defines the local coordinate system that is referenced by all geometric representations.

### Geometry Representations:

There are no attribute driven geometry representations defined for this object type in this release. Therefore, explicit geometry must be used in all cases.

### **Standard 3D Geometric Representation**

The standard geometric representation for this object type is defined using **explicit geometry**. A faceted boundary representation (faceted BRep) should be used. These can include voids (IfcFacetedBrepWithVoids) or exclude them (IfcFacetedBrep).

### **Advanced 3D Geometric Representation**

There is no advanced geometry representation defined for this object type in this release.

### **Arbitrary 3D Geometric Representation**

There is no arbitrary geometry representation defined for this object type in this release, since the standard representation can be arbitrary.

## **22.22. PropertySet Pset\_AccessoryCommon**

### **22.22.1. PropertySet Semantic Definition**

*Definition from IAI:* Set of properties common to all Built-In Accessories.

### **22.22.2. Attribute and Relationship Definitions**

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManufactureInfo	reference to Manufacturer information	IfcObjectReference	IfcGloballyUniqueId, IfcManufactureInformation	n/a	n/a	NIL
ManufacturerMaterial	Material selection - from the manufacturer's material options for this fixture type	IfcSimpleProperty	IfcString	n/a	n/a	empty string
ManufacturerColor	Color selection - from the manufacturer's color options for this fixture type	IfcSimpleProperty	IfcString	n/a	n/a	empty string
ManufacturerFinish	Finish selection - from the manufacturer's finish options for this fixture type	IfcSimpleProperty	IfcString	n/a	n/a	empty string

## **22.23. PropertySet Pset\_AccessoryDoorOrWindowHardware**

### **22.23.1. PropertySet Semantic Definition**

*Definition from IAI:* Commonly referred to as "Door hardware" and "Window hardware".

### **22.23.2. Attribute and Relationship Definitions**

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonAccessoryProperties	Nested PropertySet - that defines properties common to all types of Accessories	IfcObjectReference	IfcGloballyUniqueId, Pset_AccessoryCommon	n/a	n/a	NIL
ProjectHwGroupReference	Project reference ID for this standard collection of hardware elements for doors	IfcSimpleProperty	IfcString	see type	see type	empty string

TypeDescription	Description for this type of frame (note name is captured in the TypeDef object that references this PropertySet)	IfcSimpleProperty	IfcString	see type	see type	empty string
DoorHardwareElementList	A LIST enumeration values - referencing an IfcEnumeratedProperty that defines IfcDoorHardwareElementEnum.	IfcPropertyList	IfcEnumeratedProperty, Pset_DoorHardwareElementEnum (Hingeset, Lockset, Handset, Deadbolt, Kickplate, Pushplate, Peephole, Knocker, DoorStop, Passthrough)			

## 22.24. PropertySet Pset\_AccessoryPublicRestroom

### 22.24.1. PropertySet Semantic Definition

*Definition from IAI:* These are what are commonly referred to as "Restroom Accessories".

### 22.24.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonAccessoryProperties	Nested PropertySet - that defines properties common to all types of Accessories	IfcObjectReference	IfcGloballyUniqueld, Pset_AccessoryCommon	n/a	n/a	NIL

## 22.25. PropertySet Pset\_AccessoryUnspecified

### 22.25.1. PropertySet Semantic Definition

*Definition from IAI:* All other types of accessories (not specified in other types).

### 22.25.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonAccessoryProperties	Nested PropertySet - that defines properties common to all types of Accessories	IfcObjectReference	IfcGloballyUniqueld, Pset_AccessoryCommon	n/a	n/a	

## 22.26. PropertySet Pset\_CabinetCommon

### 22.26.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all cabinets.

### 22.26.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManuafactureInfo	reference to Manufacturer information	IfcObjectReference	IfcGloballyUniqueld, IfcManufactureInformation	n/a	n/a	NIL

## 22.27. PropertySet Pset\_CabinetOffice

### 22.27.1. PropertySet Semantic Definition

*Definition from IAI:* Cabinet designed for use in a commercial office space.

### 22.27.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonCabinetProperties	Nested PropertySet - that defines properties common to all types of Cabinets	IfcObjectReference	IfcGloballyUniqueId, Pset_CabinetCommon	n/a	n/a	NIL

## 22.28. PropertySet Pset\_CabinetRestroom

### 22.28.1. PropertySet Semantic Definition

*Definition from IAI:* Cabinet designed for use in a Restroom (Toilet).

### 22.28.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonCabinetProperties	Nested PropertySet - that defines properties common to all types of Cabinets	IfcObjectReference	IfcGloballyUniqueId, Pset_CabinetCommon	n/a	n/a	NIL

## 22.29. PropertySet Pset\_CabinetStorage

### 22.29.1. PropertySet Semantic Definition

*Definition from IAI:* Cabinet designed for storage.

### 22.29.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonCabinetProperties	Nested PropertySet - that defines properties common to all types of Cabinets	IfcObjectReference	IfcGloballyUniqueId, Pset_CabinetCommon	n/a	n/a	NIL

## 22.30. PropertySet Pset\_CabinetUnspecified

### 22.30.1. PropertySet Semantic Definition

*Definition from IAI:* All other types of cabinets (not specified in other types).

### 22.30.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
---------------	------------	---------------	-------------------	------	------	---------

CommonCabinetProperties	Nested PropertySet - that defines properties common to all types of Cabinets	IfcObjectReference	IfcGloballyUniqueId, Pset_CabinetCommon	n/a	n/a	NIL
-------------------------	--	--------------------	---	-----	-----	-----

## 22.31. PropertySet Pset\_Counter

### 22.31.1. PropertySet Semantic Definition

*Definition from IAI:* Horizontal work surface, generally on top of a built-in cabinet.

### 22.31.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonCounterOrShelfProperties	Nested PropertySet - that defines properties common to all types of counters and shelves	IfcObjectReference	IfcGloballyUniqueId, Pset_CounterOrShelfCommon	n/a	n/a	NIL

## 22.32. PropertySet Pset\_CounterOrShelfCommon

### 22.32.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all Counters and shelves.

### 22.32.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManufactureInfo	reference to Manufacturer information	IfcObjectReference	IfcGloballyUniqueId, IfcManufactureInformation	n/a	n/a	NIL

## 22.33. PropertySet Pset\_RailingBalustrade

### 22.33.1. PropertySet Semantic Definition

*Definition from IAI:* Similar to Guardrail except the location is at the edge of a floor..

### 22.33.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonRailingProperties	Nested PropertySet - that defines properties common to all types of railings	IfcObjectReference	IfcGloballyUniqueId, Pset_RailingCommon	n/a	n/a	NIL
RepeatingElements	reference to definition of repeating rail stiles - defined in a referenced Pset.	IfcObjectReference	IfcGloballyUniqueId, Pset_RepeatingElement	n/a	n/a	NIL

## 22.34. PropertySet Pset\_RailingCommon

### 22.34.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all types of railings.

### 22.34.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManufactureInfo	reference to Manufacturer information	IfcObjectReference	IfcGloballyUniqueId, IfcManufactureInformation	n/a	n/a	NIL
RailingMaterial	Index into the IfcMaterialList defined in the IfcBuildingElement supertype	IfcObjectReference	IfcInteger	1	MaterialList length	1
Height	Height to the top of the railing - from stair, landing or floor	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0

## 22.35. PropertySet Pset\_RailingGuardrail

### 22.35.1. PropertySet Semantic Definition

*Definition from IAI:* Railings designed to guard human occupants from falling off a stair, ramp or landing where there is a vertical drop at the edge of such floors/landings of 1/2 meter or more..

### 22.35.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonRailingProperties	Nested PropertySet - that defines properties common to all types of railings	IfcObjectReference	IfcGloballyUniqueId, Pset_RailingCommon	n/a	n/a	NIL
RepeatingElements	reference to definition of repeating rail stiles - defined in a referenced Pset.	IfcObjectReference	IfcGloballyUniqueId, Pset_RepeatingElement	n/a	n/a	NIL
MountedHandrail	Reference to any handrail mounted on this guardrail	IfcObjectReference	IfcGloballyUniqueId, IfcRailing	n/a	n/a	NIL

## 22.36. PropertySet Pset\_RailingHandrail

### 22.36.1. PropertySet Semantic Definition

*Definition from IAI:* Railing designed to serve as an optional structural support for loads applied by human occupants (at hand height). Generally located adjacent to ramps and stairs. Generally floor or wall mounted..

### 22.36.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonRailingProperties	Nested PropertySet - that defines properties common to all types of railings	IfcObjectReference	IfcGloballyUniqueId, Pset_RailingCommon	n/a	n/a	NIL
HandrailMaterial	Index into the IfcMaterialList	IfcSimpleProperty	IfcInteger	1	MaterialList	1

	defined in the IfcBuildingElement supertype				list length	
HandrailHeight	Height to top of handrail - from stair, landing or floor	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0
MaxDistanceFrom Wall	Distance from the wall to the handrail surface furthest from the wall. Value of 0.0 means value not set.	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0

## 22.37. PropertySet Pset\_RampCommon

### 22.37.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all types of ramps.

### 22.37.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
RampPurpose	Purpose of primary use for this ramp	IfcSimpleProperty	IfcString	n/a	n/a	empty string
HandicapAccessible	Is this ramp rated as handicap accessible - according the local building codes.	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE

## 22.38. PropertySet Pset\_RampElemented

### 22.38.1. PropertySet Semantic Definition

*Definition from IAI:* Ramp constructed using repeating elements..

### 22.38.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonRampProp erties	Nested PropertySet - that defines properties common to all types of ramps	IfcObjectReference	IfcGloballyUniqueId, Pset_RampCommon	n/a	n/a	NIL
RepeatingElements	reference to definition of repeating elements in this ramp assembly.	IfcObjectReference	IfcGloballyUniqueId, Pset_RepeatingElement	n/a	n/a	NIL

## 22.39. PropertySet Pset\_RampLayered

### 22.39.1. PropertySet Semantic Definition

*Definition from IAI:* Ramp constructed using layered elements.



## 22.39.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonRampProperties	Nested PropertySet - that defines properties common to all types of ramps	IfcObjectReference	IfcGloballyUniqueId, Pset_RampCommon	n/a	n/a	NIL

## 22.40. PropertySet Pset\_RampSolid

### 22.40.1. PropertySet Semantic Definition

*Definition from IAI:* Ramp constructed using a single layer of solid material.

## 22.40.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonRampProperties	Nested PropertySet - that defines properties common to all types of ramps	IfcObjectReference	IfcGloballyUniqueId, Pset_RampCommon	n/a	n/a	NIL

## 22.41. PropertySet Pset\_Shelf

### 22.41.1. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonCounterOrShelfProperties	Nested PropertySet - that defines properties common to all types of counters and shelves	IfcObjectReference	IfcGloballyUniqueId, Pset_CounterOrShelfCommon	n/a	n/a	NIL

## 22.42. PropertySet Pset\_SpaceProgramCirculation

### 22.42.1. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonSpaceProgramProperties	Nested PropertySet - that defines properties common to all types of SpacePrograms	IfcObjectReference	IfcGloballyUniqueId, Pset_SpaceProgramCommon	n/a	n/a	NIL
SpacesServed	List of references to the spaces served by this circulation space	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcSpace	n/a	n/a	NIL
CirculationLoad	Maximum number of occupants per minute this space must accommodate (as in escape from Fire). Zero means the value has not been calculated.	IfcSimpleProperty	IfcInteger	0	see type	0
RequiredFFETypes	Furniture, Fixtures and Equipment for this space	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcBuildingElement	n/a	n/a	empty set

## 22.43. PropertySet Pset\_SpaceProgramCommon

### 22.43.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all types of Space Programs.

### 22.43.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ProgramSpaceDescription	Description for this space in the Architectural Program (client brief)	IfcSimpleProperty	IfcString	see type	see type	empty string
RoomNumber	Number assigned to this space	IfcSimpleProperty	IfcString	see type	see type	empty string
RoomName	Name assigned to a room space	IfcSimpleProperty	IfcString	see type	see type	empty string
ProgrammedFloorArea	The floor area programmed for this space (according to client requirements)	IfcSimpleProperty	IfcAreaMeasure	0	see type	0
RequestedLocations	Location requested by client	IfcObjectReference	IfcGloballyUniqueId, IfcSpatialElement	n/a	n/a	NIL
GeneralLocationDescription	General description of location (e.g. "third floor south")	IfcSimpleProperty	IfcString	n/a	n/a	empty string
Function	How is this space to be used	IfcSimpleProperty	IfcString	n/a	n/a	empty string
SecurityRequirements	Client requirements for security	IfcSimpleProperty	IfcString	n/a	n/a	empty string
SpecialRequirements	Client special requirements	IfcSimpleProperty	IfcString	n/a	n/a	empty string
BudgetLimits	Multiple budgets for managing this space	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcCostElement	n/a	n/a	NIL
InteractWith	List of workspace interactions in which this space participates.	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcRelWorkInteraction	n/a	n/a	NIL

## 22.44. PropertySet Pset\_SpaceProgramOccupied

### 22.44.1. PropertySet Semantic Definition

*Definition from IAI:* Space program for a space to be occupied by humans.

### 22.44.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonSpaceProgramProperties	Nested PropertySet - that defines properties common to all types of SpacePrograms	IfcObjectReference	IfcGloballyUniqueId, Pset_SpaceProgramCommon	n/a	n/a	NIL
OccupiedSpaceProgramStandard	Nested PropertySet - that defines properties for any	IfcObjectReference	IfcGloballyUniqueId, Pset_SpaceProgramOccupie	n/a	n/a	NIL

	Space Standard used		dStandard			
Occupants	List of persons who will occupy this space	IfcPropertyList	IfcObjectReference, IfcPerson	n/a	n/a	NIL
OccupantOrganization	Client organization that will be charged for this space	IfcObjectReference	IfcOrganization	n/a	n/a	NIL
OccupancyTargetDate	Target date of occupancy.	IfcObjectReference	IfcDateAndTime	see type	see type	see type
BldgCodeOccupancyType	Occupancy type according to the building code for this project	IfcSimpleProperty	IfcString	n/a	n/a	empty string
ProgrammedOccupantCount	Programmed number of occupants for this space. Zero means the value has not been set.	IfcSimpleProperty	IfcInteger	0	see type	0
RequiredFFETypes	Furniture, Fixtures and Equipment for this space	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcBuildingElement	n/a	n/a	empty set
PrivacyRequirements	Client requirements for privacy	IfcSimpleProperty	IfcString	n/a	n/a	empty string
WeeklyOccupiedHours	Hours per week that this space is programmed for occupants	IfcSimpleProperty	IfcTimeMeasure	see type	see type	0

## 22.45. PropertySet Pset\_SpaceProgramOccupiedStandard

### 22.45.1. PropertySet Semantic Definition

*Definition from IAI:* Standard (repeating) program for multiple, like spaces to be occupied by humans.

### 22.45.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonSpaceProgramProperties	Nested PropertySet - that defines properties common to all types of SpacePrograms	IfcObjectReference	IfcGloballyUniqueId, Pset_SpaceProgramCommon	n/a	n/a	NIL
EmployeeType	Examples: manager, programmer, secretary, etc. IFCs don't define standardized employee types. The user should provide the company based terms for employee types.	IfcSimpleProperty	IfcString	n/a	n/a	empty string
FurnitureStyle	The style of furniture for the space or workstation designed	IfcSimpleProperty	IfcString	n/a	n/a	empty string
CostLimit	The maximum cost limit for the space such as its interiors, furniture, etc. The context of the cost is provided through IfcCostSchedule.	IfcObjectReference	IfcGloballyUniqueId, IfcCostSchedule	n/a	n/a	NIL
StandardArea	The area programmed for this space standard	IfcSimpleProperty	IfcAreaMeasure	0	see type	0
MinimumArea	Minimum area for such a	IfcSimpleProperty	IfcAreaMeasure	0	see type	0

	space					
MaximumArea	Maximum area for such a space	IfcSimpleProperty	IfcAreaMeasure	0	see type	0
StandardLength	Standard length for spaces of this type	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0
StandardWidth	Standard width for spaces of this type	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0

## 22.46. PropertySet Pset\_SpaceProgramTechnical

### 22.46.1. PropertySet Semantic Definition

*Definition from IAI:* Space program for a space designed to house building systems, equipment or maintenance elements.

### 22.46.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonSpaceProgramProperties	Nested PropertySet - that defines properties common to all types of SpacePrograms	IfcObjectReference	IfcGloballyUniqueId, Pset_SpaceProgramCommon	n/a	n/a	NIL

## 22.47. PropertySet Pset\_StairAccess

### 22.47.1. PropertySet Semantic Definition

*Definition from IAI:* Functional stair - for access to between different floor levels.

### 22.47.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonStairProperties	Nested PropertySet - that defines properties common to all types of stairs	IfcObjectReference	IfcGloballyUniqueId, Pset_StairCommon	n/a	n/a	NIL

## 22.48. PropertySet Pset\_StairCommon

### 22.48.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all types of Stairs.

### 22.48.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
StairPurpose	Purpose of primary use for this stair	IfcSimpleProperty	IfcString	n/a	n/a	empty string
FireRating	Fire survival rating = length of time the stair enclosure/assembly will	IfcSimpleProperty	IfcTimeMeasure	see type	see type	0

	survive in case of fire					
ExitStair	Is this stair counted as an exit stair in case of fire	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE

## 22.49. PropertySet Pset\_StairFire

### 22.49.1. PropertySet Semantic Definition

*Definition from IAI:* Stair designed for escape in case of fire.

### 22.49.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonStairProperties	Nested PropertySet - that defines properties common to all types of stairs	IfcObjectReference	IfcGloballyUniqueId, Pset_StairCommon	n/a	n/a	NIL

## 22.50. PropertySet Pset\_StairOrnamental

### 22.50.1. PropertySet Semantic Definition

*Definition from IAI:* Stair which is also a significant building design element.

### 22.50.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonStairProperties	Nested PropertySet - that defines properties common to all types of stairs	IfcObjectReference	IfcGloballyUniqueId, Pset_StairCommon	n/a	n/a	NIL

## 22.51. PropertySet Pset\_VisualScreenAssembly

### 22.51.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all types of Visual Screens.

### 22.51.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonScreenProperties	Nested PropertySet - that defines properties common to all types of visual screen elements	IfcObjectReference	IfcGloballyUniqueId, Pset_VisualScreenCommon	n/a	n/a	NIL
ManufactureInfo	reference to Manufacturer information	IfcObjectReference	IfcGloballyUniqueId, IfcManufactureInformation	n/a	n/a	NIL

## 22.52. PropertySet Pset\_VisualScreenCommon

### 22.52.1. PropertySet Semantic Definition

*Definition from IAI:* Set of properties common to all types of Visual Screens.

### 22.52.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ScreenElementHeight	Height of the partition panel. Value of 0.0 means property not set.	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0
ScreenElementWidth	Width of the partition panel. Value of 0.0 means property not set.	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0
ScreenElementThickness	Thickness of the partition panel. Value of 0.0 means property not set.	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0
AssembledTopOfElementHeight	Height, from finish floor, to the top of this partition panel. Value of 0.0 means property not set.	IfcSimpleProperty	IfcPositiveLengthMeasure	0	see type	0
ManufacturerMaterial	Material selection - from the manufacturer's material options for this fixture type	IfcSimpleProperty	IfcString	n/a	n/a	empty string
ManufacturerColor	Color selection - from the manufacturer's color options for this fixture type	IfcSimpleProperty	IfcString	n/a	n/a	empty string
ManufacturerFinish	Finish selection - from the manufacturer's finish options for this fixture type	IfcSimpleProperty	IfcString	n/a	n/a	empty string

## 22.53. PropertySet Pset\_VisualScreenDoorOrGate

### 22.53.1. PropertySet Semantic Definition

*Definition from IAI:* Door/Gate element in a visual screen assembly.

### 22.53.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonScreenProperties	Nested PropertySet - that defines properties common to all types of visual screen elements	IfcObjectReference	IfcGloballyUniqueId, Pset_VisualScreenCommon	n/a	n/a	NIL

## 22.54. PropertySet Pset\_VisualScreenPanel

### 22.54.1. PropertySet Semantic Definition

*Definition from IAI:* Panel element in a visual screen assembly.

## 22.54.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonScreenProperties	Nested PropertySet - that defines properties common to all types of visual screen elements	IfcObjectReference	IfcGloballyUniqueId, Pset_VisualScreenCommon	n/a	n/a	NIL

## 22.55. PropertySet Pset\_VisualScreenPost

### 22.55.1. PropertySet Semantic Definition

*Definition from IAI:* Post element in a visual screen assembly.

### 22.55.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonScreenProperties	Nested PropertySet - that defines properties common to all types of visual screen elements	IfcObjectReference	IfcGloballyUniqueId, Pset_VisualScreenCommon	n/a	n/a	NIL

## 22.56. PropertySet Pset\_VisualScreenRestroomPartition

### 22.56.1. PropertySet Semantic Definition

*Definition from IAI:* Specialization of visual screen panel -- for privacy partitions in public restrooms.

### 22.56.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonScreenElementProperties	Nested PropertySet - that defines properties common to all types of visual screen elements	IfcObjectReference	IfcGloballyUniqueId, Pset_VisualScreenCommon	n/a	n/a	NIL

## 22.57. PropertySet Pset\_VisualScreenRestroomPartitionDoor

### 22.57.1. PropertySet Semantic Definition

*Definition from IAI:* Specialization of visual screen Door/Gate -- for doors used in privacy partitions for public restrooms.

### 22.57.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonScreenElementProperties	Nested PropertySet - that defines properties common to	IfcObjectReference	IfcGloballyUniqueId, Pset_VisualScreenCommon	n/a	n/a	NIL

	all types of visual screen elements					
HingeSideLeft	Indicates the hinged side of the door - when viewed from outside the partition enclosure. TRUE=left, FALSE=right.	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	TRUE
SwingDirectionIn	Indicates whether this door swings into or out of the partition enclosure. TRUE=swings in, FALSE=swings out.	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	TRUE

## 23. IfcConstructionMgmtDomain

The IfcConstructionManagement Schema contains defined types and classes that capture concepts and data requirements for construction management processes. They, together with models defined in IfcProcessExtension and IfcProjectMangementExtension, provide a set of model elements that support typical construction management applications and their integration.

In R2.0, most of the classes included in this schema are used to represent different types of construction resources that can support both cost estimating and work planning, and their integration.

HISTORY: renamed from schema IfcCostEstimatingDomain in R1.5.1.

### 23.1. Class IfcCMDocPackage

#### 23.1.1. Class Semantic Definition

IfcCMDocPackage is a class that represents a collection of construction management related objects (e.g. construction documents) in one place. It gathers all of the related construction objects of different types as one package. It can be used to track and allocation a specific construction management object quickly. An instance of IfcCMDocPackage doesn't contain the objects directly; rather, it keeps all the references of the objects and maintain the linages to the objects. IfcCMDocPackage is a subtype of IfcControl.

#### History

New Entity in IFC Release 2.0

#### 23.1.2. Attribute and Relationship Definitions

##### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcControl
      IfcCMDocPackage

```

##### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	DocPackageID	The identifier of the construction management project package given by user.	STRING	empty string	n/a	empty string



	DocPackageName	The name of the construction management project package given by user.	STRING	empty string	n/a	empty string
OPT	Description	General description of the construction management project package.	STRING	empty string	n/a	empty string
	CreationDate	The date that the package is created.	IfcDateTimeSelect	see type	see type	see type
	Authors	The authors of the package	SET [0:?] OF IfcActorSelect	N/a	n/a	N/a
	Approvals	References to the relevant instances of IfcApproval that relate to construction management in the project	SET [0:?] OF IfcApproval	see type	see type	see type
	WorkPlans	The task schedules for the project project	SET [0:?] OF IfcWorkPlan	N/a	n/a	N/a
	CostEstimates	The cost estimates for the project project	SET [0:?] OF IfcCostSchedule	N/a	n/a	N/a
	WorkOrders	Work orders generated in the project	SET [0:?] OF IfcWorkOrder	N/a	n/a	N/a
	PurchaseOrders	Purchase orders generated in the project	SET [0:?] OF IfcPurchaseOrder	N/a	n/a	N/a
	ChangeOrders	Change orders generated in the project for change of work.	SET [0:?] OF IfcChangeOrder	N/a	n/a	N/a
	Documents	All the file documents required and generated for the project	SET [0:?] OF IfcDocumentReference	N/a	n/a	N/a
	BudgetSources		SET [0:?] OF IfcBudget	N/a	n/a	N/a

### 23.1.3. Interface Definitions

- I\_CMDocPackage

### 23.1.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation

## 23.2. Class IfcConstructionEquipmentResource

### 23.2.1. Class Semantic Definition

IfcConstructionEquipmentResource represents types of construction equipment which occurrences are used as resources in a construction process. Equipment resources are wholly or partially consumed, or occupied (i.e. used) in the performance of construction processes (i.e. IfcWorkTask).

IfcConstructionEquipmentResource is not the same as IfcEquipment; the former represents a type of construction equipment that can be used to aid in performing a work task, while the latter represents equipment pieces that are part of the building as a final product of building element.

IfcConstructionEquipmentResource is a subtype of IfcResource.

#### History

New Entity in IFC Release 2.0

### 23.2.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

IfcRoot

IfcObject  
IfcResource  
**IfcConstructionEquipmentResource**

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	EquipmentModel	The model of the type of equipment.	STRING	empty string	n/a	empty string
OPT	Manufacturer	The manufacturer that produces this type of equipment.	IfcOrganization	see type	see type	see type

## 23.2.3. Interface Definitions

- I\_ConstructionEquipmentResource

## 23.2.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 23.3. Class IfcConstructionMaterialResource

### 23.3.1. Class Semantic Definition

IfcConstructionMaterialResource represents material resource types in a construction project. Their occurrences are consumed (wholly or partially), or occupied during a construction work task (i.e. IfcWorkTask).

Similar to IfcProductResource, sometimes things such as 5 tons of gravals are already instantiated as an IfcProduct because it is a result of a work task e.g. 'transporting gravals'. In this case, the instance of resource IfcConstructionMaterialResource can be associated with the product instance '5 tons of gravals' to provide more information for resource uses. Nevertheless, IfcConstructionMaterialResource should only be used to represent resource types, e.g. 'graval', but not product substances, e.g. '5 tons of graval'. IfcConstructionMaterialResource is a subtype of IfcResource.

Note that this class is not the same as IfcMaterial; the former can typically represent the type of bulk materials such as sand, gravals, nails and so on (note these can be instantiated from IfcProduct as well depending their uses in the system) used in a construction process. The latter is about physical materials used in a physical building element typically with detailed positioning (e.g. offset) and layering information.

### History

New Entity in IFC Release 2.0

## 23.3.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

IfcRoot  
IfcObject  
IfcResource  
**IfcConstructionMaterialResource**

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
--	----------------------	------------	-------------------	------	------	---------

	Suppliers	Possible suppliers of the type of materials.	SET [0:?] OF IfcOrganization	N/a	n/a	N/a
OPT	OrderQuantity	The basic quantity for ordering.	IfcMeasureWithUnit	See type	See type	See type
	MaterialProducts	The products that are produced from other work tasks, but used as the materials.	SET [0:?] OF IfcProduct	N/a	n/a	N/a
	DesignMaterial	the materials specified by the design process that needs to be procured in the construction processes as the resource.	SET [0:?] OF IfcMaterial	N/a	n/a	N/a

### 23.3.3. Interface Definitions

- I\_ConstructionMaterialResource

### 23.3.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 23.4. Class IfcConstructionZoneAggregationProduct

### 23.4.1. Class Semantic Definition

IfcConstructionZoneAggregationProduct can be used to represent two concepts: one is a construction zone, and the other is a construction aggregation area. It is defined for use by construction cost estimating and scheduling.

A construction zone allows for the grouping of products into a single product element. A defined zone that includes multiple products (i.e. IfcProduct), to which a work task or a group of work tasks takes place or a cost estimate is calculated. In particular, where work may need to be planned to e.g. build all the columns on the 2<sup>nd</sup> floor. For its intended use (i.e. cost estimating and work planning), IfcConstructionZoneAggregationProduct should be a type of product since it must be considered to be an output result of a process that requires resources to produce this 'product'.

A construction aggregation is a defined area on a product (i.e. IfcProduct) representing a part of the product, to which a work task or a group of work tasks takes place or a cost estimate is calculated.

The construction aggregation allows for the breakdown of work into more atomic elements for costing. In particular, where work may need to be done in several parts e.g. the building of a single wall which may be done in several 'lifts' or the casting of a slab where the slab may comprise several regional casts. It should be a type of product since it must be considered to be an output result of a process that requires certain resources.

#### History

New Entity in IFC Release 2.0

### 23.4.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcConstructionZoneAggregationProduct

```

## Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	ID	The identity given to a construction zone or aggregation product.	STRING	empty string	n/a	empty string
OPT	Name	The name of the construction zone or aggregation.	STRING	empty string	n/a	empty string
OPT	Description	The description of the construction zone or aggregation.	STRING	empty string	n/a	empty string
	ZoneNotAggregation	Indicates whether the object is a zone product or aggregation product. If the value is TRUE, it means it is a zone; otherwise it is an aggregation product type.	BOOLEAN	TRUE	FALSE	TRUE
OPT	SelectionCriteria	Specification requirements applying to a construction zone or aggregation.	STRING	empty string	n/a	empty string
OPT	PartOfProduct	The reference product that the aggregation is part of.	IfcProduct	see type	see type	see type
	CoveredProducts	The references of products that the construction zone covers.	SET [0:?] OF IfcProduct	N/a	N/a	N/a

### 23.4.3. Interface Definitions

- I\_ConstructionZoneAggregationProduct

## 23.5. Class IfcCrewResource

### 23.5.1. Class Semantic Definition

IfcCrewResource represents a type of resource used in construction processes, i.e. construction crew resource. A construction crew resource typically includes labor resource, equipment resource, material resource, subcontractor resource, as well as other crew resources. Construction crew resources are partially or wholly consumed, or occupied in a construction process (i.e. IfcWorkTask). Since IfcCrewResource represents the resource types, individual persons are not required to be identified and linked to the crew resource, while they can be associated if needed. IfcCrewResource is a subtype of IfcResource.

Note that the IfcCrewResource is enabled to contain other crew resource types (i.e. crew resource type nesting) through its relationship to IfcRelCrewContainsResources. The WHERE constraints of the entity define the mechanism of how this nesting can be achieved.

Also note that in practice, when defining a crew resource type, a set of process types that the crew type is suitable for, are also identified. Since IFC R2.0 doesn't handle process types (IfcProcess represents process occurrences) explicitly, this relationship is not handled in R2.0.

Additionally, the term 'crew resource' is used as a standard term for the kind of resources described in North America. This requirement is based on the IAI project 'Cost Estimating' defined in North America Project Management Domain committee. It is acknowledged that this term may not be a popular term in the construction management industries in other countries. The class can be considered to be renamed when more projects are defined to provide such requirements.

### History

New Entity in IFC Release 2.0

## 23.5.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcResource
      IfcCrewResource
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
INV	RequiresResources	Any other resources required in the crew resource.	IfcRelAggregatesCrewResources	see type	see type	see type

## 23.5.3. Interface Definitions

- I\_CrewResource

## 23.5.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation

## 23.6. Class IfcLaborResource

### 23.6.1. Class Semantic Definition

IfcLaborResource represents labor resource types used in a construction work task (i.e. IfcWorkTask). It implies a type of labor with particular skills or crafts required to perform certain type of construction or management related work. Therefore, labor resource types typically do not identify individual persons (i.e. IfcActors) for cost estimating purpose. IfcLaborResource is a subtype of IfcResource.

### History

New Entity in IFC Release 2.0

## 23.6.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcResource
      IfcLaborResource
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	Title	The title of the type of labor such as carpenter, crane operator, superintendent, etc.	STRING	empty string	n/a	empty string
	SkillSet	The skill set required for this type of labor.	SET [0:?] OF STRING	N/a	n/a	N/a

### 23.6.3. Interface Definitions

- I\_LaborResource

### 23.6.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 23.7. Class IfcProductResource

### 23.7.1. Class Semantic Definition

IfcProductResource represents a type of construction resource, that is, product resources. Product resources are roles of products that are consumed (wholly or partially), or occupied (i.e. used) in the performance of a construction work task. Occurrences of products that are used as product resources are once instances of IfcProduct since they are resulted from some processes. For instance, formworks can be instantiated as products resulted from process 'constructing formwork'. However, they become to be used as resources in process 'pouring concrete' in a later stage of the project.

IfcProductResource is modeled as a type of resource (i.e. subtype of IfcResource) that identifies a product linked and that describes how it can be used as a resource through IfcRelResourceUse in a process.

#### History

New Entity in IFC Release 2.0

### 23.7.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcResource
      IfcProductResource
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	ResourceProduct	This identifies the product that is being used as the resource	IfcProduct	see type	see type	see type

### 23.7.3. Interface Definitions

- I\_ProductResource

### 23.7.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 23.8. Class *IfcRelAggregatesCrewResources*

### 23.8.1. Class Semantic Definition

IfcRelRelAggregatesCrewResources is class that enables a construction crew resource type (i.e. IfcCrewResource) to contain other resource types. It specifies the quantity of the included resources in the crew resource. It can also specify the conversion rate of the resource when being included in the resource. IfcRelRelAggregatesCrewResources is a subtype of ifcRelationship.

#### History

New Entity in IFC Release 2.0

### 23.8.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcRelationship
    IfcRelAggregatesCrewResources
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	RelatingCrewResource		IfcCrewResource			
	RelatedResources		LIST [1:?] OF IfcResource			
	RequiredQuantity		REAL	0	n/a	1
OPT	ConversionRate	The conversion rate for the resource when being included in the crew resource.	IfcMeasureWithUnit	see type	see type	see type

### 23.8.3. Interface Definitions

- I\_RelAggregatesCrewResources

### 23.8.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation

## 23.9. Class *IfcSubcontractResource*

### 23.9.1. Class Semantic Definition

#### History

New Entity in IFC Release 2.0

### 23.9.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  
```

IfcObject  
IfcResource  
IfcSubcontractResource

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	SubcontractResourceID	The id to identify the subcontract.	STRING	empty string	n/a	empty string
OPT	Company	The company that the subcontract is signed with.	IfcOrganization	see type	see type	see type
OPT	JobDescription	The description of the jobs that this subcontract should complete.	STRING	empty string	n/a	empty string

## 23.9.3. Interface Definitions

- I\_SubcontractResource

# 24. IfcFacilitiesMgmtDomain

The IfcFacilitiesMgmtDomain Schema defines basic concepts in the facilities management (FM) domain. This schema, along with IfcProcessExtension and IfcProjectManagementExtension, provide a set of models that can be used by typical facilities management applications.

In R2.0, these models can be used to support FM processes such as furniture and equipment scheduling, occupancy and space planning, move management, and workstation design and layout, etc. When the objects defined in these schemas are generated by these processes, their values can be made available based on IFC data structure for other FM processes to use.

HISTORY: existing schema from R1.5.1.

## 24.1. Type IfcFurnitureElementTypeEnum

### 24.1.1. Type Semantic Definition

#### History

New Enumeration in IFC Release 2.0

### 24.1.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcSystemFurnitureElement

### 24.1.3. Enumeration

Panel
Worksurface
Storage
UserDefined
NotDefined



## 24.2. *Type IfcFurnitureTypeEnum*

### 24.2.1. Type Semantic Definition

#### **History**

This Enumeration has changed after IFC Release 1.5.1, please see the Migration Guide for details

### 24.2.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcFurniture

### 24.2.3. Enumeration

Table
Chair
Desk
FileCabinet
UserDefined
NotDefined

## 24.3. *Type IfcInventoryTypeEnum*

### 24.3.1. Type Semantic Definition

#### **History**

New Enumeration in IFC Release 2.0

### 24.3.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcInventory

### 24.3.3. Enumeration

AssetInventory
SpaceInventory
UserDefined
NotDefined

## 24.4. *Type IfcOccupancyMoveTypeEnum*

### 24.4.1. Type Semantic Definition

#### **History**

New Enumeration in IFC Release 2.0

## 24.4.2. Enumeration

MoveIn
MoveOut
MoveInOut
UserDefined
NotDefined

## 24.5. Type *IfcWorkstationEnum*

### 24.5.1. Type Semantic Definition

#### **History**

New Enumeration in IFC Release 2.0

### 24.5.2. Enumeration

Workstation
Workstationgroup
UserDefined
NotDefined

## 24.6. Class *IfcFurniture*

### 24.6.1. Class Semantic Definition

It represents a piece of furniture (e.g. office furniture such as table, desk, chair, file cabinet etc). *IfcFurniture* represents the type of furniture that is not usually fixed to the building or assembled as system or modular furniture such as workstations (i.e. office cubes) as well as their components such as panels, work surfaces, etc.

ISSUE: New attributes and relationships are required for IFC R2.0 as shown in the following table (existing attributes and relationships are not shown.)

The data type of *AssignedTo* is changed from 'Ref. *IfcActor*' to 'SET [0:?] Ref. *IfcActorSelect*'.

#### **History**

This Entity has changed after IFC Release 1.5.1, please see the Migration Guide for details

### 24.6.2. Attribute and Relationship Definitions

#### **Superclasses and Subclasses**

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcFurniture

```

## Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	4 predefined generic types are possible. Use Type Definition corresponding to this generic type	IfcFurnitureTypeEnum	Chair	Table	Chair
	AssignedTo	Persons, Departments, Organizations to which this piece of furniture is assigned	SET [0:?] OF IfcActorSelect	N/a	N/a	N/a
	FurnitureModel		IfcFurnitureModel	see type	see type	see type

## Formal Propositions

WR61	
------	--

## 24.6.3. Interface Definitions

- I\_Furniture

## 24.6.4. Type Definitions

### Common PropertySet

Pset\_FurnitureCommon

### Type driven PropertySets

PreDefined Type	Associated PropertySet
Table	Pset_Table
Chair	Pset_Chair
Desk	Pset_Desk
FileCabinet	Pset_FileCabinet
UserDefined	
NotDefined	

## 24.6.5. Geometry Use Definitions

Geometric use of IfcFurniture follows that of IfcProduct.

## 24.7. Class IfcFurnitureModel

### 24.7.1. Class Semantic Definition

This class represents features captured from the feature listing of a 'type' or 'model' of furniture defined in a furniture catalog of a furniture manufacturer. The features described through the values of the attributes of IfcFurnitureModel are furniture features that generally apply to all the specific pieces of furniture of the model. These features are manufacturer-dependent and thus must be provided by the manufacturer that makes and/or supplies the furniture. The use of IfcFurnitureModel to IfcFurniture is in form of a reference, that is, an instance of IfcFurnitureModel in a computer system should be referenced (e.g. through the instance id, or memory pointer) by one or more pieces of instances of the furniture of the same model. Note that this class can enable the direct linkage between the furniture and the furniture model data handled within the manufacturer's catalog in the manufacturer's computer systems, if these systems are IFC compliant and understand the semantics of IfcFurnitureModel. This class is a subtype of IfcControl.

## History

New Entity in IFC Release 2.0

## 24.7.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcControl
      IfcFurnitureModel
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	ModelID	An id for the model given by the manufacturer	STRING	Empty string	N/a	Empty string
	ModelName	A textual short description of the name of the model	STRING	Empty string	N/a	Empty string
	Manufacturer	The manufacturer that provides the furniture model and that makes the pieces of furniture referring to the model.	IfcOrganization	see type	see type	see type
OPT	CatalogName	The name of the catalog that the model belongs to and that maintained by the manufacturer	STRING	Empty string	N/a	Empty string
OPT	BasicDescription	The basic description of the model provided by the manufacturer	STRING	Empty string	N/a	Empty string
	BasicFeatures	The list of features in textual form provided by the manufacturer	LIST [0:?] OF STRING	N/a	N/a	N/a
	AdjustableFeatures	The list of adjustable features in textual form provided by the manufacturer. An adjustable feature is a function that allows some part of furniture to be adjustable to better meet the use needs, e.g. adjustable arms of a chair.	LIST [0:?] OF STRING	N/a	N/a	N/a
	Options	The list of options that the model comes with in textual form provided by the manufacturer. An option can be in terms of colors, sizes, etc.	LIST [0:?] OF STRING	N/a	N/a	N/a
OPT	MaintenanceManual	A reference to a document (e.g. document name), a file (e.g. file name), or a manufacturer defined id referring to the version of the manual.	STRING	Empty string	N/a	Empty string
OPT	WarrantyDetails	A reference to a document (e.g. document name), a file (e.g. file name), or a manufacturer defined id referring to the version of the warranty policy document.	STRING	Empty string	N/a	Empty string

## 24.7.3. Interface Definitions

- I\_FurnitureModel

## 24.7.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.8. Class IfcInventory

### 24.8.1. Class Semantic Definition

IfcInventory represents information about an inventory – data about a collection of items for an enterprise. Two types of inventory are handled in this release-the space inventory and asset inventory (i.e. inventory for furniture, fixture and equipment). The links between the assets and spaces are handled through the object relationships of the IfcSpace, IfcFurniture, and IfcEquipment. IfcInventory is a subtype of IfcGroup.

#### History

New Entity in IFC Release 2.0

### 24.8.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcGroup
      IfcInventory
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	The type of inventory such as space or asset, with enum value of SpaceInventory, or AssetInventory, respectively	IfcInventoryTypeEnum	AssetInventory	SpaceInventory	AssetInventory
	Jurisdiction	The organizational unit of the inventory	IfcOrganization	see type	see type	see type
	Responsible	Persons who are responsible for the inventory	SET [0:?] OF IfcActorSelect	N/a	N/a	N/a
	LastUpdateDate	The date of last update	IfcDateTimeSelect	see type	see type	see type
	CurrentValue	An estimate cost value of the inventory. Cost contexts such as re-sell value are provided through IfcCostSchedule.	SET [0:?] OF IfcCostElement	N/a	N/a	N/a
	OriginalValue	Original cost value of the total inventory. Cost contexts such as purchase costs, installation costs, etc. are provided through IfcCostSchedule.	SET [0:?] OF IfcCostElement	N/a	N/a	N/a

#### Formal Propositions

WR41	
------	--

### 24.8.3. Interface Definitions

- I\_Inventory

## 24.8.4. Type Definitions

### *Type driven PropertySets*

PreDefined Type	Associated PropertySet
AssetInventory	Pset_AssetInventory
SpaceInventory	Pset_SpaceInventory
UserDefined	
NotDefined	

## 24.8.5. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.9. Class IfcOccupancySchedule

### 24.9.1. Class Semantic Definition

IfcOccupancySchedule represents a plan for activities to move people and FF&E from spaces to spaces. IfcOccupancySchedule contains a list of elements (i.e. IfcOccupancyScheduleElement) linked with an occupancy task with their logical relationships. This class also contains references to a set of spaces to be occupied, emptied, and re-occupied. An instance of IfcOccupancySchedule can also contain other schedule instances through IfcRelNestsOccupancySchedules. IfcOccupancySchedule is a subtype of IfcControl.

### *History*

New Entity in IFC Release 2.0

### 24.9.2. Attribute and Relationship Definitions

#### *Superclasses and Subclasses*

```

IfcRoot
  IfcObject
    IfcControl
      IfcOccupancySchedule
  
```

#### *Attributes and Relationships*

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	SpaceProgramsToMove	All the space programs to be moved in this plan	SET [0:?] OF IfcSpaceProgram	N/a	N/a	N/a
	NewlyOccupiedSpaces	Spaces that are occupied in this plan	SET [0:?] OF IfcSpace	N/a	N/a	N/a
	NewlyEmptiedSpaces	Spaces that are emptied in this plan	SET [0:?] OF IfcSpace	N/a	N/a	N/a
	ReoccupiedSpaces	Spaces that are occupied by new tenants in this plan	SET [0:?] OF IfcSpace	N/a	N/a	N/a
	OccupantsToMove	People or organizations that are moving out or in the spaces	SET [0:?] OF IfcActorSelect	N/a	N/a	N/a
	ScheduleElements	The occupancy activities involved in this plan	SET [0:?] OF IfcOccupancyScheduleElement	N/a	N/a	N/a

### *Formal Propositions*

WR41	Containing processes are IfcOccupancyActivity
------	---

WR42	
------	--

### 24.9.3. Interface Definitions

- I\_OccupancySchedule

### 24.9.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.10. Class IfcOccupancyScheduleElement

### 24.10.1. Class Semantic Definition

IfcOccupancyScheduleElement is an occupancy task element in an occupancy schedule (i.e. IfcOccupancySchedule). It represents an occupancy task by linking an IfcOccupancyTask instance providing task time control information through IfcScheduleTimeControl defined in IfcProcessExtension schema. An instance of IfcOccupancyScheduleElement can include other elements through IfcRelNestsOccupancyScheduleElements. IfcOccupancyScheduleElement is a subtype of IfcControl.

#### History

New Entity in IFC Release 2.0

### 24.10.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcControl
      IfcOccupancyScheduleElement

```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	OccupancyTask	The work task that the schedule element assigned to.	IfcOccupancyTask	See type	See type	See type
	TimeForSchedule		IfcScheduleTimeControl	See type	See type	See type

#### Formal Propositions

WR41	
WR42	

### 24.10.3. Interface Definitions

- I\_OccupancyScheduleElement

### 24.10.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.11. Class IfcOccupancyTask

### 24.11.1. Class Semantic Definition

IfcOccupancyTask represents an activity to move people around office spaces along with furniture and equipment, etc. It is represented by IfcOccupancyScheduleElement that provides time control data in an occupancy plan (i.e. IfcOccupancySchedule). An instance of IfcOccupancyTask can also contain other instances of the same type through the nesting capability provided by IfcProcess. IfcOccupancyTask is a subtype of IfcProcess.

#### History

New Entity in IFC Release 2.0

### 24.11.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProcess
      IfcOccupancyTask
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	OccupantsToMove	People or organizations that are moving out or in the spaces	SET [0:?] OF IfcActorSelect	N/a	N/a	N/a
	FFEtomove	The furniture, fixture and equipment that are moved out or in the spaces	SET [0:?] OF IfcBuildingElement	N/a	N/a	N/a
OPT	MoveType	Indicates whether this occupancy activity is to move in to space(s), or out from space(s), or to involve both. The relations in 'MoveFrom' and 'MoveTo' must be set by the user to ensure consistency. That is: if MoveIn, 'MoveFrom' should be empty; if MoveOut, 'MoveIn' should be empty; if Both, neither should be empty.	IfcOccupancyMoveTypeEnum	MoveIn	MoveOut	MoveIn
	MoveFrom	The spaces from which people or FF&E are moving out from.	SET [0:?] OF IfcSpace	N/a	N/a	N/a
	MoveTo	The spaces to which people or FF&E are moving into.	SET [0:?] OF IfcSpace	N/a	N/a	N/a
OPT	ConstraintType	The activity constraint for timing. The type of the constraint such as 'as soon as possible', 'not start before', 'must start on', 'must finish on', 'must start before', 'must finish before', 'may start after', 'must move out by', etc. The value of 'ConstraintType' and 'ConstraintDate' makes a meaningful constraint.	STRING	empty string	n/a	empty string
OPT	ConstraintTime	the date requirement for certain constrainttype such as must move out by 'date', etc.	IfcDateTimeSelect	see type	see type	see type
INV	ScheduleElements	The work schedule elements that associates with this work tasks.	SET [0:?] OF IfcOccupancyScheduleElement	See type	See type	See type



			nt			
--	--	--	----	--	--	--

### Formal Propositions

WR41	
------	--

## 24.11.3. Interface Definitions

- I\_OccupancyTask

## 24.11.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.12. Class IfcRelNestsOccupancyScheduleElements

### 24.12.1. Class Semantic Definition

IfcRelNestsOccupancyScheduleElements is an objectified relationship enabling a mechanism to allow one occupancy schedule element to include other elements of the same type.  
IfcRelNestsOccupancyScheduleElements is a subtype of IfcRelNests.

#### History

New Entity in IFC Release 2.0

## 24.12.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcRelationship
    IfcRelNests
      IfcRelNestsOccupancyScheduleElements

```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	Description	Any description that would be useful to understand the nesting of the schedules.	STRING	Empty string	N/a	Empty string

### Formal Propositions

WR41	Nesting object must be of type IfcOccupancyScheduleElement.
WR42	Nesting objects must be of type IfcOccupancyScheduleElement.

## 24.12.3. Interface Definitions

- I\_RelNestsOccupancyScheduleElements

## 24.12.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.13. Class *IfcRelNestsOccupancySchedules*

### 24.13.1. Class Semantic Definition

IfcRelNestsOccupancySchedules is an objectified relationship enabling a mechanism to allow one occupancy schedule to include other schedules of the same type. IfcRelNestsOccupancySchedules is a subtype of IfcRelNests.

#### History

New Entity in IFC Release 2.0

### 24.13.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcRelationship
    IfcRelNests
      IfcRelNestsOccupancySchedules
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
OPT	Description	Any description that would be useful to understand the nesting of the schedules.	STRING	Empty string	N/a	Empty string

#### Formal Propositions

WR41	Nesting object must be of type IfcWorktaskSchedule.
WR42	Nesting objects must be of type IfcWorktaskSchedule.

### 24.13.3. Interface Definitions

- I\_RelNestsOccupancySchedules

### 24.13.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.14. Class *IfcRelWorkInteraction*

### 24.14.1. Class Semantic Definition

IfcRelWorkInteraction is an objectified relationship. It represents an interaction relationship between two parties, such as employees, organizations, or space programs (interaction relations between workstations, workstation groups, floor blocks, or spaces, are defined through their associated space programs). The work interaction relationship can be used for defining the adjacency between space programs as well as spaces. In work interaction instance can be created by an architectural space design program or an facilities management space planning program. By storing the work intereaction instances in the central model of the building project supported by IFCs, it can be reused when spaces need to be re-designed, re-modeled, or people need to move offices, etc. IfcRelWorkInteraction is a subtype of IfcRelationship.

## History

New Entity in IFC Release 2.0

## 24.14.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

IfcRoot  
IfcRelationship  
IfcRelWorkInteraction

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	RelatingActor	One of the two parties involved in the interaction. 'RelatingActor' and 'RelatedActor' must be two different parties.	IfcActor	see type	see type	see type
	RelatedActor	One of the two parties involved in the interaction. 'RelatingActor' and 'RelatedActor' must be two different parties.	IfcActor	see type	see type	see type
OPT	RelatingSpaceProgram	One of the two space programs involved in the interaction. If RelatingActor or RelatedActor is also used, the user is responsible for make sure the the space program referenced is consistent with that associated by RelatingActor. RelatingSpaceProgram and RelatedSpaceProgram must refer to two different instances of IfcSpaceProgram.	IfcSpaceProgram	see type	see type	see type
OPT	RelatedSpaceProgram	See RelatingSpaceProgram.	IfcSpaceProgram	see type	see type	see type
OPT	Description	General description of the interaction	STRING	empty sting	n/a	empty sting
OPT	DailyFrequency	Number of interactions daily	INTEGER	0	see type	0
OPT	ImportanceRating	Represents the level of importance of interaction	INTEGER	0	see type	0
OPT	AverageDuration	Average time duration of each interaction	IfcTimeMeasure	see type	see type	see type
OPT	Location	The location where this interaction happens.	IfcSpace	see type	see type	see type

## 24.14.3. Interface Definitions

- I\_WorkInteraction

## 24.14.4. Geometry Use Definitions

Instances of this class have no physical presence and therefore no geometric representation.

## 24.15. Class *IfcSystemFurnitureElement*

### 24.15.1. Class Semantic Definition

This class represents a component (i.e. modular element) of systems furniture (i.e. modular furniture) such as a vertical panel, a work surface, and storage which must be used in assembly of a system furniture unit such as a workstation or workstation group. *IfcSystemFurnitureElement* doesn't provide the functions for people to use as a type of furniture, but it is typically used in assembling the systems furniture as an integrated part. *IfcSystemFurnitureElement* is a subtype of *IfcBuildingElement*.

#### History

New Entity in IFC Release 2.0

### 24.15.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcSystemFurnitureElement

```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Panel, Worksurface, Storage.	<i>IfcFurnitureElementTypeEnum</i>	Panel	Storage	Panel
INV	ElementOf	The workstation that this element is used in.	SET [0:?] OF <i>IfcWorkstation</i>	n/a	n/a	N/a

#### Formal Propositions

WR61	
------	--

### 24.15.3. Interface Definitions

- *I\_SystemFurnitureElement*

### 24.15.4. Type Definitions

#### Type driven PropertySets

PreDefined Type	Associated PropertySet
Panel	Pset_Panel
Worksurface	Pset_Worksurface
Storage	Pset_Storage
NotDefined	
UserDefined	

### 24.15.5. Geometry Use Definitions

Geometric use of *IfcSystemFurnitureElement* follows *IfcBuildingElement*.

## 24.16. Class IfcWorkstation

### 24.16.1. Class Semantic Definition

A unit of an office working area that is bounded by modular vertical panels and that is assembled by a set of modular system furniture elements such as work surfaces, file storage, etc. It can also have free stand furniture such as desks, chairs, and bookshelves. Office equipment such as fixture lighting and computers, printers, etc can be included in a workstation through IfcRelContains with ContainmentType=SpaceContainer of IfcObject. A workstation can be considered to be a spatial element, i.e. a space, since it is a place that provides an office space; it however also has characteristics of a type of furniture since it is made by a furniture manufacturer as a product. In IFCs, IfcWorkstation is modeled as a subtype of IfcSpace with properties matched to a manufactured furniture product using Pset\_FurnitureCommon property set that can be attached through IfcExtensionPropertySet at IfcObject level.

In IFCs, a workstation group is also modeled by IfcWorkstation that consists of a set of workstations connected with modular vertical panels. A workstation group is a workplace and can be physically viewed as a bigger and complex workstation that is usually for a group of people who work, in the individual workstations, as a functional team or a departmental unit. A workstation group can contain FF&E that don't belong to any individual workstations and are shared by all the personnel and workstations within the group. Like normal workstation, a workstation group also possesses both spatial and furniture characteristics.

#### History

New Entity in IFC Release 2.0

### 24.16.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcSpatialElement
        IfcSpace
          IfcWorkstation
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	WorkstationType	To indicate whether the workstation represents a single office cube or a workstation group which includes a set of cubes.	IfcWorkstationEnum	Workstation	Workstationgroup	Workstation
	FurnitureModel	To reference an instance of IfcFurnitureModel that represents a list of features or options that apply to this workstation provided by the manufacturer.	IfcFurnitureModel	See type	See type	See type
	ModuleElements	list of worksurfaces and storage, excluding the vertical panels.	SET [0:?] OF IfcSystemFurnitureElement	n/a	n/a	n/a
OPT	TotalWorkTaskZone	The total value of work task area within the workstation	IfcAreaMeasure	See type	See type	See type
OPT	TotalChairClearence	The total value of chair clearence area within the workstation	IfcAreaMeasure	See type	See type	See type
OPT	TotalCirculation	The total value of circulation area within the workstation	IfcAreaMeasure	See type	See type	See type

OPT	TotalCubes	If the number is greater than 0, it indicates that the workstation is a workstation group. The number indicates the total number of the workstations contained in the workstation group.	INTEGER	0	n/a	0
-----	------------	--	---------	---	-----	---

### Informal Propositions

IP1	FurnitureElementType of IfcSystemFurnitureElement of ModuleElements must be either Worksurface or Storage, but not panels
IP2	Vertical Panels that bound the workstaion are referenced through 'BoundedBy' of IfcSpace
IP3	The workstatiion group that assembles the workstation is referenececd in the 'PartOfAssembly' of IfcSpace

## 24.16.3. Interface Definitions

- I\_Workstation

## 24.16.4. Geometry Use Definitions

Geometric use of IfcWorkstation follows that of IfcSpace.

## 24.17. PropertySet Pset\_AssetInventory

### 24.17.1. PropertySet Semantic Definition

A set of specific properties for asset (i.e FF&E) inventory.

### 24.17.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
TotalOriginalValue	the original total value of all the assets	IfcObjectReference	IfcCost	see type	see type	see type
TotalCurrentValue	the current total value of all the assets	IfcObjectReference	IfcCost	see type	see type	see type
TotalItems	total items in the inventory	IfcSimpleProperty	IfcInteger	0	n/a	0

## 24.18. PropertySet Pset\_Chair

### 24.18.1. PropertySet Semantic Definition

A set of specific properties for furniture type chair.

### 24.18.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonFurnitureProperties	Reference to a nested SharedPropertySet, containing properties common to all types of furniture objects.	IfcObjectReference	IfcGloballyUniqueid, Pset_FurnitureCommon	see type	see type	see type

SeatingHeight	The value of seating height if the chair height is not adjustable.	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
HighestSeatingHeight	The value of seating height of high level if the chair height is adjustable.	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
LowestSeatingHeight	The value of seating height of low level if the chair height is adjustable.	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type

## 24.19. PropertySet Pset\_Desk

### 24.19.1. PropertySet Semantic Definition

A set of specific properties for furniture type desk.

### 24.19.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonFurnitureProperties	Reference to a nested SharedPropertySet, containing properties common to all types of furniture objects.	IfcObjectReference	IfcGloballyUniqueId, Pset_FurnitureCommon	see type	see type	see type
WorksurfaceArea	The value of the work surface area of the desk.	IfcSimpleProperty	IfcAreaMeasure	see type	see type	see type

## 24.20. PropertySet Pset\_FileCabinet

### 24.20.1. PropertySet Semantic Definition

A set of specific properties for furniture type file cabinet.

### 24.20.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonFurnitureProperties	Reference to a nested SharedPropertySet, containing properties common to all types of furniture objects.	IfcObjectReference	IfcGloballyUniqueId, Pset_FurnitureCommon	see type	see type	see type
WithLock	Indicates whether the file cabinet is lockable or not.	IfcSimpleProperty	IfcBoolean	Yes	No	Yes

## 24.21. PropertySet Pset\_FurnitureCommon

### 24.21.1. PropertySet Semantic Definition

Common properties for all types of furniture such as chair, desk, table, and file cabinet.

## 24.21.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
Description	Specific description of this type of furniture.	IfcSimpleProperty	IfcString	empty string	n/a	empty string
AssetInformation	Information about this asset. A String containing the name of the IfcOccurrencePropertySet - Pset_Asset. Empty string means that this information not yet set.	IfcObjectReference	IfcGloballyUniqueid, Pset_Asset	empty string	n/a	empty string
ManufactureInformation	Reference to a SharedPropertySet (Pset_ManufactureInformation) containing information about the manufacture of this furniture type.	IfcObjectReference	IfcGloballyUniqueid, Pset_ManufactureInformation	see type	see type	see type
ElementMaintenance	Information about the condition and maintenance of this furniture. A object reference to the OccurrencePropertySet ("Pset_ElementMaintenance"). This Pset will be attached to the subject object - in the list of OccurrencePropertysets defined in the IfcObject supertype. An empty reference means that this information is not yet set.	IfcObjectReference	IfcGloballyUniqueid, Pset_ElementMaintenance	empty string	n/a	empty string
Style	Description of the furniture style	IfcSimpleProperty	IfcString	empty string	n/a	empty string
Height	The nominal height of the furniture of this type	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
Length	The nominal length or width of the furniture of this type	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
Depth	The nominal depth of the furniture of this type	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
MainColor	The main color of the furniture of this type	IfcSimpleProperty	IfcString	empty string	n/a	empty string
Material	the main material the furniture of this type is made of, e.g. walnut, etc.	IfcObjectReference	IfcGloballyUniqueid, Pset_MaterialSet	see type	see type	see type
Finishing	e.g. walnut, fabric	IfcObjectReference	IfcGloballyUniqueid, Pset_MaterialSet	see type	see type	see type



## 24.22. *PropertySet Pset\_FurnitureElementCommon*

### 24.22.1. PropertySet Semantic Definition

Common properties for all systems furniture (i.e. modular furniture) element types (e.g. vertical panels, work surfaces, and storage).

### 24.22.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
IsUsed	indicates whether the element is being used in a workstation or not.	IfcSimpleProperty	IfcBoolean	Yes	No	Yes
GroupCode	e.g. panels, worksurfaces, storage, etc.	IfcSimpleProperty	IfcString	empty	n/a	empty string
Width	i.e. nominal width	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
Height	i.e. nominal length	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
Finishing	e.g. walnut, fabric	IfcSimpleProperty	IfcString	empty string	n/a	empty string

## 24.23. *PropertySet Pset\_Panel*

### 24.23.1. PropertySet Semantic Definition

A set of specific properties for vertical panels that assembly workstations.

### 24.23.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonProperties	The common property values shared by many panel instances.	IfcObjectReference	IfcGloballyUniqueId, Pset_FurnitureElementCommon	see type	see type	see type
Shape	the vertical boundary shape of the panel	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcProductDefinitionShape	see type	see type	see type
HasOpening	indicates whether the panel has an opening or not.	IfcSimpleProperty	IfcBoolean	Yes	No	Yes
PanelType	e.g. Acoustical, Horz_Seg, Monolithic, Glazed, Open, Ends, Door, Screen, etc.	IfcSimpleProperty	IfcString	empty string	n/a	empty string
Thickness	the thickness of the panel	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type

## 24.24. *PropertySet Pset\_SpaceInventory*

### 24.24.1. PropertySet Semantic Definition

A set of specific properties for space inventory.

## 24.24.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
TotalSpaces	total number of spaces in the inventory	IfcSimpleProperty	IfcInteger	0	n/a	0
TotalNetArea	total net area of all the spaces; can be calculated from each space	IfcSimpleProperty	IfcAreaMeasure	see type	see type	see type

## 24.25. PropertySet Pset\_Storage

### 24.25.1. PropertySet Semantic Definition

A set of specific properties for storage used in workstations.

### 24.25.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonProperties	The common property values shared by many storage instances.	IfcObjectReference	IfcGloballyUniqueId, Pset_FurnitureElementCommon	see type	see type	see type
IsOverhead	is overhead storage or not	IfcSimpleProperty	IfcBoolean	YES	NO	YES
SupportType	i.e. Freestanding or supported	IfcSimpleProperty	IfcString	empty string	n/a	empty string
UsePurpose	e.g. shelf, stationary, office supplies, personal items, etc.	IfcSimpleProperty	IfcString	empty string	n/a	empty string
NumberOfDrawers	number of drawers	IfcSimpleProperty	IfcInteger	0	n	0
HangingHeight	hanging height if IsOverhead	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
Depth	depth of the storage	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type

## 24.26. PropertySet Pset\_Table

### 24.26.1. PropertySet Semantic Definition

A set of specific properties for furniture type table.

### 24.26.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonFurnitureProperties	Reference to a nested SharedPropertySet, containing properties common to all types of furniture objects.	IfcObjectReference	IfcGloballyUniqueId, Pset_FurnitureCommon	n/a	n/a	n/a
WorksurfaceArea	The value of the work surface area of the desk.	IfcSimpleProperty	IfcAreaMeasure	see type	see type	see type
NumberOfChairs	Maximum number of chairs that can fit with the table for normal use.	IfcSimpleProperty	IfcInteger	1	n/a	1

## 24.27. PropertySet Pset\_Worksurface

### 24.27.1. PropertySet Semantic Definition

A set of specific properties for work surfaces used in workstations.

### 24.27.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
CommonProperties	The common property values shared by many work surface instances.	IfcObjectReference	IfcGloballyUniqueId, Pset_FurnitureElementCommon	see type	see type	see type
UsePurpose	e.g. writing/reading, computer, meeting, printer, reference files, etc.	IfcSimpleProperty	IfcString	empty string	n/a	empty string
SupportType	i.e. Freestanding or supported	IfcSimpleProperty	IfcString	empty string	n/a	empty string
HangingHeight	the hanging height of the worksurface	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
Thickness	the thickness of the worksurface	IfcSimpleProperty	IfcPositiveLengthMeasure	see type	see type	see type
ShapeDescription	corner square, rectangle, etc.	IfcSimpleProperty	IfcString	empty string	n/a	empty string

## 25. IfcHvacDomain

The IfcHvacDomain schema in the domain layer defines basic object concepts required for interoperability between Building Service domain extensions (notably HVAC) and other domain extensions defined in the current IFC model. This schema is new in IFC R2.0.

### 25.1. Type IfcActuatorFailPositionEnum

#### 25.1.1. Type Semantic Definition

*Definition from IAI:* This enumeration defines the different states for failure for an instance of IfcActuator.

#### History

New Enumeration in IFC Release 2.0

#### 25.1.2. Enumeration

FailOpen
FailClosed
UserDefined
NotDefined

## 25.2. Type *IfcActuatorTypeEnum*

### 25.2.1. Type Semantic Definition

*Definition from IAI:* This enumeration defines the different types of discrete elements an IfcActuator object can fulfill.

#### **History**

New Enumeration in IFC Release 2.0

### 25.2.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcActuator

### 25.2.3. Enumeration

ElectricActuator
PneumaticActuator
HydraulicActuator
HandOperatedActuator
UserDefined
NotDefined

## 25.3. Type *IfcAirTerminalBoxTypeEnum*

### 25.3.1. Type Semantic Definition

*Definition from IAI:* This enumeration identifies different types of air terminal boxes. Note that this enumeration does not define type: it is informational only.

ISSUES: See I-472 for changes made in IFC Release 2.0 Beta 3

#### **History**

New Enumeration in IFC Release 2.0

### 25.3.2. Enumeration

VariableAirVolume
ConstantVolume
VariableAirVolumeReheat
ConstantVolumeReheat
VariableAirVolumeDualDuct
ConstantVolumeDualDuct
FanPowered
UserDefined
NotDefined

## 25.4. Type *IfcControllerTypeEnum*

### 25.4.1. Type Semantic Definition

*Definition from IAI:* This enumeration defines the different types of discrete elements an IfcController object can fulfill.

#### **History**

New Enumeration in IFC Release 2.0

### 25.4.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcController

### 25.4.3. Enumeration

HvacController
UserDefined
NotDefined

## 25.5. Type *IfcDamperSizingMethodEnum*

### 25.5.1. Type Semantic Definition

*Definition from IAI:* This enumeration defines the sizing methods used for an instance of IfcDamper.

#### **History**

New Enumeration in IFC Release 2.0

### 25.5.2. Enumeration

Nominal
Exact
UserDefined
NotDefined

## 25.6. Type *IfcDamperTypeEnum*

### 25.6.1. Type Semantic Definition

*Definition from IAI:* This enumeration defines the different types of discrete elements an IfcDamper object can fulfill.

#### **History**

New Enumeration in IFC Release 2.0

## 25.6.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcDamper

## 25.6.3. Enumeration

FireDamper
SmokeDamper
FireSmokeDamper
BackdraftDamper
ControlDamper
Louver
UserDefined
NotDefined

## 25.7. Type IfcSensorTypeEnum

### 25.7.1. Type Semantic Definition

*Definition from IAI:* This enumeration defines the different types of discrete elements an IfcSensor object can fulfill.

#### **History**

New Enumeration in IFC Release 2.0

### 25.7.2. PreDefined Type

This enumeration defines the available PreDefined Types for IfcSensor

## 25.7.3. Enumeration

HvacSensor
UserDefined
NotDefined

## 25.8. Type IfcValveEnum

### 25.8.1. Type Semantic Definition

*Definition from IAI:* This enumeration identifies different types of valves. Note that this enumeration does not define type: it is informational only.

#### **History**

New Enumeration in IFC Release 2.0

## 25.8.2. Enumeration

Automated
Angle

AWWA
Ball
Butterfly
Check
Diverter
Gate
Globe
Nace
Needle
Plug
Pipeline
Safety
Threeway
ULFM
UserDefined
NotDefined

## 25.9. Class IfcActuator

### 25.9.1. Class Semantic Definition

*Definition from IA1:* This class defines properties of an actuating device typically used in a control system such as a building automation control system.

#### History

New Entity in IFC Release 2.0

### 25.9.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcDistributionElement
            IfcDistributionControlElement
              IfcActuator

```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Predefined generic types are specified in an Enumeration. A TypeDefinition is available for each generic type (as the required attributes differ). Use Type Definition corresponding to this generic type.	IfcActuatorTypeEnum	ElectricActuator	HandOperatedActuator	ElectricActuator
	FailPosition	Enumeration defining the types of fail positions for the actuator	IfcActuatorFailPositionEnum	FailOpen	Unset	FailOpen

## Formal Propositions

WR81	
------	--

### 25.9.3. Interface Definitions

- I\_Actuator

### 25.9.4. Type Definitions

#### Type driven PropertySets

PreDefined Type	Associated PropertySet
ElectricActuator	Pset_ElectricActuator
PneumaticActuator	Pset_PneumaticActuator
HydraulicActuator	Pset_HydraulicActuator
HandOperatedActuator	Pset_HandOperatedActuator
UserDefined	
NotDefined	

### 25.9.5. Geometry Use Definitions

#### Object Geometry in Context

The geometric representation of IfcActuator is given by the IfcProductShape, allowing multiple geometric representations. Included are:

#### Local Position

The local placement for IfcActuator is defined in its supertype, IfcProduct. It is defined by the

- IfcLocalPlacement, which defines the local coordinate system that is referenced by all geometric representations.

#### Standard Geometric Representation

The standard geometric representation of IfcActuator is defined using **explicit geometry**. The faceted B-Rep capabilities (with or without voids) shall be supported for standard representation.

Currently, the usage of attribute driven geometry for IfcActuator is not supported.

## 25.10. Class IfcAirTerminalBox

### 25.10.1. Class Semantic Definition

*Definition from IA1:* This class defines properties of an air terminal box, which participates in an HVAC duct distribution system. An air terminal box is typically used to control or modulate the amount of air delivered to its downstream ductwork.

#### History

New Entity in IFC Release 2.0



## 25.10.2. Attribute and Relationship Definitions

### *Superclasses and Subclasses*

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcDistributionElement
            IfcDistributionFlowElement
              IfcFlowController
                IfcAirTerminalBox

```

### *Attributes and Relationships*

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	TerminalBoxType	Enumeration that identifies the type of terminal box (e.g., VariableAirVolume, ConstantVolume, VariableAirVolumeReheat, ConstantVolumeReheat, FanPowered, VariableAirVolumeDualDuct, etc.)	IfcAirTerminalBoxTypeEnum	Variable AirVolum e	Unset	Variable AirVolum e
	SoundLevel	Design sound power level	STRING	see type	see type	empty string

## 25.10.3. Interface Definitions

- I\_AirTerminalBox

## 25.10.4. Geometry Use Definitions

### *Object Geometry in Context*

The geometric representation of IfcAirTerminalBox is given by the IfcProductShape, allowing multiple geometric representations. Included are:

### *Local Position*

The local placement for IfcAirTerminalBox is defined in its supertype, IfcProduct. It is defined by the

- IfcLocalPlacement, which defines the local coordinate system that is referenced by all geometric representations.

### *Standard Geometric Representation*

The standard geometric representation of IfcAirTerminalBox is defined using **explicit geometry**. The faceted B-Rep capabilities (with or without voids) shall be supported for standard representation.

Currently, the usage of attribute driven geometry for IfcAirTerminalBox is not supported.

## 25.11. Class IfcController

### 25.11.1. Class Semantic Definition

*Definition from IAI:* This class defines properties of a controller which interacts with other devices in a control system such as a building automation control system.

## History

New Entity in IFC Release 2.0

## 25.11.2. Attribute and Relationship Definitions

### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcDistributionElement
            IfcDistributionControlElement
              IfcController
  
```

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Predefined generic types are specified in an Enumeration. A TypeDefinition is available for each generic type (as the required attributes differ). Use Type Definition corresponding to this generic type.	IfcControllerTypeEnum	HvacCon troller	HvacCon troller	HvacCon troller

### Formal Propositions

WR81	
------	--

## 25.11.3. Interface Definitions

- I\_Controller

## 25.11.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
HvacController	Pset_HvacController
UserDefined	
NotDefined	

## 25.11.5. Geometry Use Definitions

### Object Geometry in Context

The geometric representation of IfcController is given by the IfcProductShape, allowing multiple geometric representations. Included are:

### Local Position

The local placement for IfcController is defined in its supertype, IfcProduct. It is defined by the

- IfcLocalPlacement, which defines the local coordinate system that is referenced by all geometric representations.

## Standard Geometric Representation

The standard geometric representation of IfcController is defined using **explicit geometry**. The faceted B-Rep capabilities (with or without voids) shall be supported for standard representation.

Currently, the usage of attribute driven geometry for IfcController is not supported.

## 25.12. Class IfcDamper

### 25.12.1. Class Semantic Definition

*Definition from IAI:* This class defines elements of a damper, which typically is used in an HVAC air distribution system to control or modulate the flow of air.

#### History

New Entity in IFC Release 2.0

### 25.12.2. Attribute and Relationship Definitions

#### Superclasses and Subclasses

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcDistributionElement
            IfcDistributionFlowElement
              IfcFlowController
                IfcDamper
  
```

#### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Predefined generic types are specified in an Enumeration. A TypeDefinition is available for each generic type (as the required attributes differ). Use Type Definition corresponding to this generic type.	IfcDamperTypeEnum	FireDamper	Louver	FireDamper
	FrameDepth	The length (or depth) of the damper frame	IfcLengthMeasure	see type	see type	0
	SizingMethod	Enumeration that identifies whether the damper is sized nominally or with exact measurements.	IfcDamperSizingMethodEnum	Nominal	Exact	Nominal
	CloseOffRating	Close off rating - IfcMeasureWithUnit (IfcPressureMeasure)	IfcMeasureWithUnit	see type	see type	0
	LeakageAirFlowrate	Leakage air flow rate - IfcMeasureWithUnit (IfcVolumetricFlowrateMeasure)	IfcMeasureWithUnit	see type	see type	0

#### Formal Propositions

WR91	
------	--

## 25.12.3. Interface Definitions

- I\_Damper

## 25.12.4. Type Definitions

### *Type driven PropertySets*

PreDefined Type	Associated PropertySet
FireDamper	Pset_FireDamper
SmokeDamper	Pset_SmokeDamper
FireSmokeDamper	Pset_FireSmokeDamper
BackdraftDamper	Pset_BackdraftDamper
ControlDamper	Pset_ControlDamper
Louver	Pset_Louver
UserDefined	
NotDefined	

## 25.12.5. Geometry Use Definitions

### *Object Geometry in Context*

The geometric representation of IfcDamper is given by the IfcProductShape, allowing multiple geometric representations. Included are:

### *Local Position*

The local placement for IfcDamper is defined in its supertype, IfcProduct. It is defined by the

- IfcLocalPlacement, which defines the local coordinate system that is referenced by all geometric representations.

### *Standard Geometric Representation*

The standard geometric representation of IfcDamper is defined using **explicit geometry**. The faceted B-Rep capabilities (with or without voids) shall be supported for standard representation.

Currently, the usage of attribute driven geometry for IfcDamper is not supported.

## 25.13. Class IfcSensor

### 25.13.1. Class Semantic Definition

*Definition from IAI:* This class defines properties of a sensor which is used for detection in a control system such as a building automation control system.

### *History*

New Entity in IFC Release 2.0

### 25.13.2. Attribute and Relationship Definitions

#### *Superclasses and Subclasses*

IfcRoot  
IfcObject

IfcProduct  
IfcElement  
IfcBuildingElement  
IfcDistributionElement  
IfcDistributionControlElement  
**IfcSensor**

### Attributes and Relationships

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	PredefinedType	Predefined generic types are specified in an Enumeration. A TypeDefinition is available for each generic type (as the required attributes differ). Use Type Definition corresponding to this generic type.	IfcSensorTypeEnum	HvacSensor	HvacSensor	HvacSensor

### Formal Propositions

WR81	
------	--

## 25.13.3. Interface Definitions

- I\_Sensor

## 25.13.4. Type Definitions

### Type driven PropertySets

PreDefined Type	Associated PropertySet
HvacSensor	Pset_HvacSensor
UserDefined	
NotDefined	

## 25.13.5. Geometry Use Definitions

### Object Geometry in Context

The geometric representation of IfcSensor is given by the IfcProductShape, allowing multiple geometric representations. Included are:

### Local Position

The local placement for IfcSensor is defined in its supertype, IfcProduct. It is defined by the

- IfcLocalPlacement, which defines the local coordinate system that is referenced by all geometric representations.

### Standard Geometric Representation

The standard geometric representation of IfcSensor is defined using **explicit geometry**. The faceted B-Rep capabilities (with or without voids) shall be supported for standard representation.

Currently, the usage of attribute driven geometry for IfcSensor is not supported.

## 25.14. Class *IfcValve*

### 25.14.1. Class Semantic Definition

*Definition from IA1:* This class defines elements of a valve, which typically is used in an HVAC piping distribution system to control or modulate the flow of the fluid.

#### **History**

New Entity in IFC Release 2.0

### 25.14.2. Attribute and Relationship Definitions

#### **Superclasses and Subclasses**

```

IfcRoot
  IfcObject
    IfcProduct
      IfcElement
        IfcBuildingElement
          IfcDistributionElement
            IfcDistributionFlowElement
              IfcFlowController
                IfcValve
  
```

#### **Attributes and Relationships**

	Attribute / Relation	Definition	Data or Rel. Type	Min.	Max.	Default
	CloseOffRating	Close off rating	IfcMeasureWithUnit	see type	see type	0
	ValveCv	Cv value for the valve	REAL	see type	see type	0
	ValveType	Type of valve	IfcValveEnum	Automated	Unset	Gate

### 25.14.3. Interface Definitions

- I\_Valve

### 25.14.4. Geometry Use Definitions

#### **Object Geometry in Context**

The geometric representation of IfcValve is given by the IfcProductShape, allowing multiple geometric representations. Included are:

#### **Local Position**

The local placement for IfcValve is defined in its supertype, IfcProduct. It is defined by the

- IfcLocalPlacement, which defines the local coordinate system that is referenced by all geometric representations.

#### **Standard Geometric Representation**

The standard geometric representation of IfcValve is defined using **explicit geometry**. The faceted B-Rep capabilities (with or without voids) shall be supported for standard representation.

Currently, the usage of attribute driven geometry for IfcValve is not supported.

## 25.15. PropertySet Pset\_AnalogInput

### 25.15.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of an analog input.

### 25.15.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
HighLimit	The high limit value for the analog input.	IfcSimplePropertyWithUnit	IfcReal, Unspecified	see type	see type	0
LowLimit	The low limit value for the analog input.	IfcSimplePropertyWithUnit	IfcReal, Unspecified	see type	see type	0
Deadband	The deadband value for the analog input.	IfcSimplePropertyWithUnit	IfcReal, Unspecified	see type	see type	0
HighLimitEnable	Is HighLimit validation enabled (TRUE) or not (FALSE).	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE
LowLimitEnable	Is LowLimit validation enabled (TRUE) or not (FALSE).	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE
EventEnable	Enumeration that defines the type of event enabling	IfcEnumeratedProperty	Pset_EventEnableEnum(To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			
NotifyTypeEnum	Enumeration that defines the notification type	IfcEnumeratedProperty	Pset_NotifyTypeEnum(Alarm, Event, AcknowledgeNotification, Other, NotKnown, Unset)			

## 25.16. PropertySet Pset\_AnalogOutput

### 25.16.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of an analog output.

### 25.16.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
HighLimit	The high limit value for the analog output.	IfcSimplePropertyWithUnit	IfcReal, Unspecified	see type	see type	0
LowLimit	The low limit value for the analog output.	IfcSimplePropertyWithUnit	IfcReal, Unspecified	see type	see type	0
Deadband	The deadband value for the analog output.	IfcSimplePropertyWithUnit	IfcReal, Unspecified	see type	see type	0
HighLimitEnable	Is HighLimit validation enabled (TRUE) or not (FALSE).	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE
LowLimitEnable	Is LowLimit validation enabled (TRUE) or not (FALSE).	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE
EventEnable	Enumeration that defines the type of event enabling	IfcEnumeratedProperty	Pset_EventEnableEnum(To-OffNormal, To-Fault, To-			

			Normal, Other, NotKnown, Unset)			
NotifyTypeEnum	Enumeration that defines the notification type	IfcEnumeratedProperty	Pset_NotifyTypeEnum(Alarm, Event, AcknowledgeNotification, Other, NotKnown, Unset)			

## 25.17. PropertySet Pset\_BackdraftDamper

### 25.17.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a backdraft damper.

### 25.17.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
FrameType	The type of frame used by the damper (e.g., Standard, Single Flange, Single Reversed Flange, Double Flange, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
Actuator	Actuator references an IfcActuator object which contains the actuator information, if an actuator is part of the damper assembly	IfcObjectReference	IfcGloballyUniqueId, IfcActuator	n/a	n/a	NIL

## 25.18. PropertySet Pset\_BinaryInput

### 25.18.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a binary input.

### 25.18.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
Polarity	Enumeration defining the polarity	IfcEnumeratedProperty	Pset_PolarityEnum(Normal, Reverse, Other, NotKnown, Unset)			
InactiveText	String value to be displayed in an inactive, off, or idle state	IfcSimpleProperty	IfcString	see type	see type	empty string
ActiveText	String value to be displayed in an active, on, or running state	IfcSimpleProperty	IfcString	see type	see type	empty string
AlarmValue	Enumeration defining the operating state of the control system element	IfcEnumeratedProperty	Pset_AlarmValueEnum(Inactive, Active, Other, NotKnown, Unset)			
EventEnable	Enumeration that defines the type of event enabling	IfcEnumeratedProperty	Pset_EventEnableEnum(To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			
AckedTransitions	Enumeration that defines the	IfcEnumeratedProperty	Pset_AckedTransitionsEnum(			



	type of transition acknowledgement	erty	To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			
--	------------------------------------	------	--	--	--	--

## 25.19. PropertySet Pset\_BinaryOutput

### 25.19.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a binary output.

### 25.19.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
Polarity	Enumeration defining the polarity	IfcEnumeratedProperty	Pset_PolarityEnum(Normal, Reverse, Other, NotKnown, Unset)			
InactiveText	String value to be displayed in an inactive, off, or idle state	IfcSimpleProperty	IfcString	see type	see type	empty string
ActiveText	String value to be displayed in an active, on, or running state	IfcSimpleProperty	IfcString	see type	see type	empty string
MinimumOffTime	Minimum Off Time	IfcObjectReference	IfcLocalTime	see type	see type	0
MinimumOnTime	Minimum On Time	IfcObjectReference	IfcLocalTime	see type	see type	0
FeedbackValue	Enumeration defining the feedback value from the control system element	IfcEnumeratedProperty	Pset_FeedbackValueEnum(In active, Active, Other, NotKnown, Unset)			
EventEnable	Enumeration that defines the type of event enabling	IfcEnumeratedProperty	Pset_EventEnableEnum(To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			
AckedTransitions	Enumeration that defines the type of transition acknowledgement	IfcEnumeratedProperty	Pset_AckedTransitionsEnum(To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			

## 25.20. PropertySet Pset\_ControlDamper

### 25.20.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a control damper.

### 25.20.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
DesignAirVelocity	The design air velocity for the damper assembly	IfcSimplePropertyWithUnit	IfcReal, LinearVelocityUnit	see type	see type	0
BladeAction	Enumeration that identifies the blade closing action for the damper	IfcEnumeratedProperty	Pset_DamperBladeActionEnum(Parallel, Opposed, Other, NotKnown, Unset)			
BladeType	The type of blade used in the damper (e.g., Triple Vee, Fabricated Airfoil, Extruded	IfcSimpleProperty	IfcString	see type	see type	empty string

	Airfoil, etc.)					
BladeMaterial	The primary material used to construct the damper blade	IfcObjectReference	IfcMaterial	n/a	n/a	NIL
BladeThickness	The thickness of the damper blade	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
FrameType	The type of frame used by the damper (e.g., Standard, Single Flange, Single Reversed Flange, Double Flange, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
FrameMaterial	The primary material used to construct the damper frame	IfcObjectReference	IfcMaterial	n/a	n/a	NIL
FrameThickness	The thickness of the damper frame	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
Actuator	Actuator references an IfcActuator object set which contains the actuator information, if an actuator is part of the damper assembly	IfcObjectReference	IfcGloballyUniqueId, IfcActuator	n/a	n/a	NIL

## 25.21. PropertySet Pset\_ElectricActuator

### 25.21.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of an electric actuator.

### 25.21.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManualOverride	Identifies whether hand-operated operation is provided as an override	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE
InputPower	Maximum input power requirement	IfcSimplePropertyWithUnit	IfcReal, PowerUnit	see type	see type	0

## 25.22. PropertySet Pset\_FireDamper

### 25.22.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a fire damper.

### 25.22.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
BladeType	Enumeration that identifies the different types of blades in the fire damper	IfcEnumeratedProperty	Pset_FireDamperBladeTypeEnum(ParallelBlade, FoldingCurtain, Other, NotKnown, Unset)			
ActuationType	Enumeration that identifies the different types of dampers	IfcEnumeratedProperty	Pset_FireDamperActuationTypeEnum(Gravity, Spring, Other, NotKnown, Unset)			

ClosureRating	Enumeration that identifies the closure rating for the damper	IfcEnumeratedProperty	Pset_FireDamperClosureRatingEnum(Dynamic, Static, Other, NotKnown, Unset)			
FireResistanceRating	Measure of the fire resistance rating in hours (e.g., 1.5 hours, 2 hours, etc.).	IfcSimpleProperty	IfcReal	0	see type	2
MountingPosition	Enumeration that identifies how the damper is mounted in the building	IfcEnumeratedProperty	Pset_DamperMountingPositionEnum(Horizontal, Vertical, Other, NotKnown, Unset)			
FusibleLinkTemperature	The temperature that the fusible link melts	IfcSimpleProperty	IfcThermodynamicTemperatureMeasure	see type	see type	0
SleeveLength	The length of the damper sleeve	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
SleeveThickness	The thickness of the damper sleeve	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
DamperLocationInSleeve	The location within the sleeve where the damper is mounted (e.g., Center)	IfcSimpleProperty	IfcString	see type	see type	empty string

## 25.23. PropertySet Pset\_FireSmokeDamper

### 25.23.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a combination fire and smoke damper.

### 25.23.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
FrameThickness	The thickness of the damper frame	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
FireResistanceRating	Measure of the fire resistance rating in hours (e.g., 1.5 hours, 2 hours, etc.).	IfcSimpleProperty	IfcReal	0	see type	2
BladeType	The type of blade used in the damper (e.g., Triple Vee, Fabricated Airfoil, Extruded Airfoil, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
MountingPosition	Enumeration that identifies how the damper is mounted in the building	IfcEnumeratedProperty	Pset_DamperMountingPositionEnum(Horizontal, Vertical, Other, NotKnown, Unset)			
FusibleLinkTemperature	The temperature that the fusible link melts	IfcSimpleProperty	IfcThermodynamicTemperatureMeasure	see type	see type	0
ControlType	The type of control used to operate the damper (e.g., Open/Closed Indicator, Resettable Temperature Sensor, Temperature Override, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
SleeveLength	The length of the damper sleeve	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
SleeveThickness	The thickness of the damper sleeve	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0

DamperLocationInSleeve	The location within the sleeve where the damper is mounted (e.g., Center)	IfcSimpleProperty	IfcString	see type	see type	empty string
Actuator	Actuator references an IfcActuator object which contains the actuator information, if an actuator is part of the damper assembly	IfcObjectReference	IfcGloballyUniqueId, IfcActuator	n/a	n/a	NIL

## 25.24. PropertySet Pset\_HandOperatedActuator

### 25.24.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a hand-operated actuator.

### 25.24.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManualOverride	Identifies whether hand-operated operation is provided as an override	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	TRUE

## 25.25. PropertySet Pset\_HvacController

### 25.25.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the general characteristics of a controller used in an HVAC control system.

### 25.25.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
Sensors	IfcSensor objects related to the controller	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcSensor	n/a	n/a	NIL
Actuators	IfcActuator objects related to the controller	IfcPropertyList	IfcObjectReference, IfcGloballyUniqueId, IfcActuator	n/a	n/a	NIL

## 25.26. PropertySet Pset\_HvacSensor

### 25.26.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the general characteristics of a sensor used in an HVAC control system.

### 25.26.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
---------------	------------	---------------	-------------------	------	------	---------

SensorType	Enumeration that identifies the type of HVAC sensor	IfcEnumeratedProperty	Pset_SensorTypeEnum(Flow, Pressure, Temperature, Gas, Concentration, Volts, Amps, Density, Viscosity, Energy, Humidity, Other, NotKnown, Unset)			
Range	The range of the sensor	IfcSimpleProperty	IfcReal	see type	see type	0
Accuracy	The accuracy of the sensor	IfcSimpleProperty	IfcReal	see type	see type	0

## 25.27. PropertySet Pset\_HydraulicActuator

### 25.27.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a hydraulic actuator.

### 25.27.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManualOverride	Identifies whether hand-operated operation is provided as an override		IfcBoolean	FALSE	TRUE	FALSE
InputPressure	Maximum design pressure for the actuator	IfcSimplePropertyWithUnit	IfcReal, PressureUnit	see type	see type	0
InputFlowrate	Maximum hydraulic flowrate requirement	IfcSimplePropertyWithUnit	IfcReal, VolumetricFlowrateUnit	see type	see type	0

## 25.28. PropertySet Pset\_LinearActuator

### 25.28.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a linear actuator.

### 25.28.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
FailDirection	Enumeration that identifies the behavior of the actuator in case of power failure	IfcEnumeratedProperty	Pset_LinearActuatorFailDirectionEnum(FailIn, FailOut, Other, NotKnown, Unset)			
Force	Indicates the maximum close-off force for the actuator	IfcSimplePropertyWithUnit	IfcReal, ForceMeasure	see type	see type	0
Stroke	Indicates the maximum distance the actuator must traverse	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0

## 25.29. PropertySet Pset\_Louver

### 25.29.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a louver.

## 25.29.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
FrameType	The type of frame used by the louver (e.g., Standard, Drainable, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
FrameThickness	The thickness of the louver frame	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
BladeType	The type of blade used in the louver (e.g., "J", "K", Cheveron, Sightproof, Drainable, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
BladeThickness	The thickness of the louver blade	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
ScreenType	The type of screen used in the louver (e.g., Birdscreen, Insect Screen, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
Actuator	Actuator references an IfcActuator object which contains the actuator information, if an actuator is part of the louver assembly	IfcObjectReference	IfcGloballyUniqueld, IfcActuator	n/a	n/a	NIL

## 25.30. PropertySet Pset\_MultiStateInput

### 25.30.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a multi-state input.

### 25.30.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
NumberOfStates	Number of states for the MultiState Input	IfcSimpleProperty	IfcInteger	see type	see type	0
StateText	String value to identify the state condition. Upper limit of the list is equal to the NumberOfStates.	IfcPropertyList	IfcSimpleProperty, IfcString	see type	see type	empty string
AlarmValues	Specifies any states the present value must equal before an EventEnable shall occur. Upper limit of the list is equal to the NumberOfStates.	IfcPropertyList	IfcSimpleProperty, IfcInteger	see type	see type	0
EventEnable	Enumeration that defines the type of event enabling	IfcEnumeratedProperty	Pset_EventEnableEnum(To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			
NotifyTypeEnum	Enumeration that defines the notification type	IfcEnumeratedProperty	Pset_NotifyTypeEnum(Alarm, Event, AcknowledgeNotification, Other, NotKnown, Unset)			

## 25.31. PropertySet Pset\_MultiStateOutput

### 25.31.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a multi-state output.

### 25.31.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
NumberOfStates	Number of states for the MultiState Output	IfcSimpleProperty	IfcInteger	see type	see type	0
StateText	String value to identify the state condition. Upper limit of the list is equal to the NumberOfStates.	IfcPropertyList	IfcSimpleProperty, IfcString	see type	see type	empty string
EventEnable	Enumeration that defines the type of event enabling	IfcEnumeratedProperty	Pset_EventEnableEnum(To-OffNormal, To-Fault, To-Normal, Other, NotKnown, Unset)			
NotifyTypeEnum	Enumeration that defines the notification type	IfcEnumeratedProperty	Pset_NotifyTypeEnum(Alarm, Event, AcknowledgeNotification, Other, NotKnown, Unset)			

## 25.32. PropertySet Pset\_PneumaticActuator

### 25.32.1. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
ManualOverride	Identifies whether hand-operated operation is provided as an override	IfcSimpleProperty	IfcBoolean	FALSE	TRUE	FALSE
InputPressure	Maximum input control air pressure requirement	IfcSimplePropertyWithUnit	IfcReal, PressureUnit	see type	see type	0
InputFlowrate	Maximum input control air flowrate requirement	IfcSimplePropertyWithUnit	IfcReal, VolumetricFlowrateUnit	see type	see type	0

## 25.33. PropertySet Pset\_RotationalActuator

### 25.33.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a rotational actuator.

### 25.33.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
FailDirection	Enumeration that identifies the behavior of the actuator in case of power failure	IfcEnumeratedProperty	Pset_RotationalActuatorFailDirectionEnum(FailClockwise, FailCounterClockwise, Other, NotKnown, Unset)			

Torque	Indicates the maximum close-off torque for the actuator	IfcSimplePropertyWithUnit	IfcReal, TorqueMeasure	see type	see type	0
Range	Indicates the maximum rotation the actuator must traverse	IfcSimpleProperty	IfcPlaneAngleMeasure	see type	see type	0

## 25.34. PropertySet Pset\_SmokeDamper

### 25.34.1. PropertySet Semantic Definition

*Definition from IAI:* This property set is used to define the characteristics of a smoke damper.

### 25.34.2. Attribute and Relationship Definitions

Property Name	Definition	Property Type	Data or Rel. Type	Min.	Max.	Default
FrameThickness	The thickness of the damper frame	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
BladeType	The type of blade used in the damper (e.g., Triple Vee, Fabricated Airfoil, Extruded Airfoil, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
MountingPosition	Enumeration that identifies how the damper is mounted in the building	IfcEnumeratedProperty	Pset_DamperMountingPositionEnum(Horizontal, Vertical, Other, NotKnown, Unset)			
ControlType	The type of control used to operate the damper (e.g., Open/Closed Indicator, Resettable Temperature Sensor, Temperature Override, etc.)	IfcSimpleProperty	IfcString	see type	see type	empty string
SleeveLength	The length of the damper sleeve	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
SleeveThickness	The thickness of the damper sleeve	IfcSimpleProperty	IfcLengthMeasure	see type	see type	0
DamperLocationInSleeve	The location within the sleeve where the damper is mounted (e.g., Center)	IfcSimpleProperty	IfcString	see type	see type	empty string
Actuator	Actuator references an IfcActuator object which contains the actuator information, if an actuator is part of the damper assembly	IfcObjectReference	IfcGloballyUniqueId, IfcActuator	n/a	n/a	NIL